CURSO DE REDES E LINUX

Agosto de 1999

Núcleo Universitário do IEEE na FEUP
Comissão de Curso LEIC 2003
Reino Linux (grupo de utilizadores de Linux)

Conteúdo

1 Introdução **7**
- 1.1 Introdução ao curso . . . 7
- 1.2 O que é o Linux? . . . 7
- 1.3 Por outras palavras... . . . 8
- 1.4 Entretenimento . . . 9
- 1.5 E quanto ao meu sistema antigo... . . . 9
- 1.6 Distribuições . . . 9
- 1.7 Documentação . . . 9

2 Ambiente gráfico **11**
- 2.1 Entrar na máquina . . . 11
- 2.2 Gestores de Janelas . . . 12
  - 2.2.1 KDE . . . 12
  - 2.2.2 ICEwm . . . 14
  - 2.2.3 Window Maker . . . 16
  - 2.2.4 Fvwm . . . 18
- 2.3 Edição e Visualização de ficheiros . . . 20
  - 2.3.1 HTML . . . 20
  - 2.3.2 PostScript . . . 22
  - 2.3.3 PDF . . . 23
  - 2.3.4 DVI . . . 25
  - 2.3.5 Ficheiros de texto . . . 25
  - 2.3.6 Texto formatado . . . 26
  - 2.3.7 Imagens . . . 27

3 Comandos Unix **33**
- 3.1 Consolas ou terminais . . . 33
- 3.2 Alguns comandos básicos . . . 34
  - 3.2.1 O sistema de ficheiros . . . 34
- 3.3 Processos . . . 39
- 3.4 Mais informações . . . 40

4 Sistema Distribuído **43**
4.1 Contas dos Utilizadores 44
4.2 Sistema de Ficheiros Distribuídos 46
4.2.1 Mount 46
4.2.2 NFS 46
4.3 Serviço de Impressão Distribuída 49
4.3.1 Imprimir em Linux 49
4.4 Correio Electrónico 50
4.4.1 Leitores de Correio 51
4.5 Serviços informáticos do CICA 51
4.5.1 Acesso ao correio electrónico do CICA 52
4.5.2 O SiFEUP 53
4.6 Acesso remoto à FEUPNet 54

5 Processamento de texto **57**
5.1 Documento tipo 58
5.2 Processamento de ficheiros LaTeX 60
5.3 Outros comandos do LaTeX 62
5.4 Gráficos vectoriais usando `xfig` 66
5.5 Outras fontes de informação 68

A Como foi feito este documento **69**

B Glossário de Informática e Redes **73**

Agradecimentos

Para a realização deste curso foi necessária a participação de inúmeras pessoas às quais queriamos agradecer:

- Jaime Villate
- Rui Araújo
- Raul Oliveira
- Pedro Braga
- Nuno Dantas
- Luis Ferreira
- Comissão LEEC 2002
- Diana Soares
- Joaquim Ribeiro
- todos os formadores

Queriamos agradecer ainda às seguintes instituições pelo seu apoio e disponibilização de material e espaço:

- Departamento de Engenharia Electrotécnica e Computadores da FEUP
- IBCOM

Capítulo 1

Introdução

1.1 Introdução ao curso

Bem vindo ao curso de «Introdução a Redes e Linux» da FEUP. Este curso tem como objectivo a introdução aos normais recursos que esta faculdade proporciona, nomeadamente a FeupNET e o acesso exterior à Internet, bem como a ambientalização aos diversos sistemas operativos e utilitários, como ferramenta primordial para o sucesso escolar.

Um dos nossos focos de interesse será a rede interna desta casa, a qual, intrinseca nas suas ramificações, e interessante naquilo que nos coloca ao dispor, irá ser abordada usando um sistema operativo que todos os dias ganha novos adeptos: o Linux.

Como seria de esperar, é impossível em seis horas ensinar aquilo que nos leva anos para dominar. Sendo assim, tentaremos, de uma forma sucinta e clara, resumir todos os pontos em questão, possibilitando aos novos alunos uma mais rápida familiarização. Nao é, no entanto, objectivo fulcral o aprofundamento das mesmas. Pelo contrário, apelamos e incentivamos o interesse e a pesquisa individual, qualidades inquestionáveis na aprendizagem dos objectivos deste curso, e na vida académica no decorrer da licenciatura desta Faculdade.

Esperamos assim que todos os alunos vejam este nosso esforço como uma oferta, e em simultâneo, uma iniciativa para o melhor aproveitamento das ferramentas que todos dispomos, enquanto mostramos uma nova dimensão, perspectiva e filosofia, possivelmente fugindo àquilo com que a maioria de nos está habituado a lidar.

1.2 O que é o Linux?

Em poucas palavras, Linux é um «clone» do sistema Unix (mas completamente escrito do nada), livre (no sentido em que o seu código fonte é livre e licenciado sobre os termos da *GNU General Public License* (GPL) e multi-plataforma. Foi

originalmente escrito por Linus Torvalds em 1991 e é agora desenvolvido por um grande grupo de programadores em todo o mundo, aberto a contribuições de qualquer voluntário. Apresenta capacidades nativas de multi-processamento, redes (no sentido de *networking*), segurança e intercomunicação entre processos (em grande parte conformante com normas tais como POSIX-1, BSD e System V) e possui ainda a habilidade de correr qualquer programa *standard* de Unix com mínimas (ou até mesmo nenhumas) modificações.

Muitas das utilidades fornecidas com a maioria das distribuições de Linux, bem como outras largamente disponíveis na Internet, são também licenciadas sobre os termos da GPL. Isto conduz a um sistema robusto, aberto, completo e sofisticado, com problemas mínimos de segurança, e alta performance, quando comparado com outros sistemas operativos comerciais.

O aumento da sua popularidade coloca-o entre os sistemas operativos mais usados em todo o mundo, tornando-o num possível lider a médio prazo. O facto de o próprio comércio se estar a re-orientar, o incrível número de utilizadores existentes, e a sua fiabilidade, é o suficiente para finalmente se criar toda uma nova dimensão.

1.3 Por outras palavras...

Encarando os factos, umas das mais importantes diferenças entre o Linux e os outros sistemas operativos é a portabilidade, e o facto de se poder ver e modificar todo o seu código fonte (desde que se respeite a GPL, cujo principal objectivo é evitar que alguem tome posse do código fonte), fazendo dele um sistema tecnologicamente de ponta.

A nível de aplicativos existe, neste momento, um número incrivelmente grande e vasto em todas as áreas. Processadores de texto, folhas de cálculo, aplicações para a Internet, programas para investigação matemática e científica, multimedia e entretenimento. Em média, cinquenta novas aplicações e *updates* são lançados diariamente, e a maior parte deles são gratuitos.

Por outro lado, o Linux (tal como o Unix) é um sistema de rede desde a sua nascença. A segurança e o suporte de multi-utilizadores não são apenas mais uma capacidade, mas sim parte integrante do sistema. Esta é razão suficiente para muitos servidores correrem sobre Unix/Linux.

Do ponto de vista empresarial, além de ser mais barato do que ter de comprar, digamos, 100 ou 200 licenças de um sistema comercial, é também altamente fácil construir *clusters* de alta performance e fiabilidade. Se tomarmos em conta que algumas empresas possibilitam também suporte técnico, estamos perante um sistema, acima de tudo, rentável.

1.4 Entretenimento

A verdade é que o Linux é mais um sistema técnico e profissional do que propriamente uma boa «consola de jogos», como alguns sistemas comerciais. No entanto, esta barreira atenua dia-a-dia com os novos lançamentos de *drivers* e interfaces gráficos (tal como o `XFree86`, tendo em conta a versão 4 brevemente disponível), tornando-o inquestionavelmente tão bom no entretenimento quanto ele é nas suas aplicações mais «sérias».

1.5 E quanto ao meu sistema antigo...

Existem duas formas de preservar o seu sistema operativo antigo, e até mesmo usá-lo se for necessário. A forma mais simples e eficaz é instalar um pequeno programa, que normalmente é distríbuido, que questiona o utilizador sobre o sistema operativo a usar quando o sistema arranca. Isto permite-lhe assim ter o seu sistema antigo intacto numa partição diferente, sem ter o mínimo de problemas (os outros sistemas operativos nem saberão que o Linux está instalado!).

A outra forma é usando um emulador ou simulador. Podemos mencionar o `Wine`, um simulador que permite correr algumas aplicações do sistema Windows, ou até mesmo o `VMWare` (comercial) que virtualiza completamente uma máquina, permitindo assim correr qualquer sistema operativo sobre ela.

1.6 Distribuições

Existem várias boas distribuições de Linux, tendo cada uma as suas vantagens e desvantagens. Algumas apostam em trazer inúmeras aplicações, outras em sendo minimalistas. Algumas gostam de automatização (para novos utilizadores), outras mais orientadas para veteranos no assunto. Mas no fim de tudo, todas se baseam no mesmo núcleo (o *kernel*!). Cabe a cada utilizador decidir qual gosta mais...

Eis aqui uma pequena lista das distribuições mais conhecidas (em nenhuma ordem particular): SuSE Linux, SlackWare, Debian, RedHat, Caldera...

1.7 Documentação

Normalmente todas as distribuições trazem documentação (pelo menos a suficiente para a instalar). No entanto, existem as últimas versões de *HOWTO's*[1] e documentação variada em diversos *sites* na Internet. Um deles é o *Linux Documentation Project*, o qual possui livros inteiros e artigos em formato electrónico (Postscript, PDF, HTML, puro Texto e fontes LaTeX) abrangendo diversos assuntos com respeito ao Linux e suas aplicações, desde um simples «Manual de Instalação», até

[1] Tutoriais sobre instalação de hardware, configuração de inúmeros programas, etc.

ao «Guia do Programador». Como pode verificar, o Linux não prima pela falta de documentação.

Na FEUP tem inúmeros locais aos quais se pode dirigir para tirar dúvidas sobre Linux. O Reino Linux é um grupo de utilizadores de Linux da FEUP que pode ajudar-lhe em qualquer questão e ao qual se pode dirigir ou através da página - `http://reinolinux.fe.up.pt` - ou indo à sala «Museu» (antigo CR2) no edifício cor de rosa. Alternativamente pode contactar membros da LEIC 2003 ou qualquer formador do curso.

Muitas das dúvidas que possa ter podem já ter resposta na página do DEEC - `http://deec00ws29.fe.up.pt`. Por isso antes de ir falar com alguém passe por esta página. Aqui também vão ser disponibilizados, se necessário, ficheiros de configuração e indicações sobre como utilizar os recursos do departamento (como imprimir, como montar disquetes ou unidades zip, etc).

Capítulo 2

Ambiente gráfico

Tradicionalmente, o Unix associa-se a uma série de comandos com nomes difíceis de lembrar que são usados num terminal de texto. Essa foi a forma tradicional como o Unix foi desenvolvido, e continua a ser o modo de operação mais rápido e preferido por pessoas com muita experiência, mas existe também o modo gráfico usando o X que resulta mais fácil e agradável de usar para quem se está a iniciar no sistema operativo.

Num ambiente gráfico como o X são disponibilizadas áreas de trabalho virtuais. A gestão dessas áreas de trabalho está a cargo do gestor de janelas. Normalmente as várias áreas são configuráveis independentemente, podendo cada uma, por exemplo, apresentar uma imagem de fundo diferente.

Uma outra peculiariedade do sistema de janelas X, para quem não esteja familiarizado com ele, é que os três botões do rato (esquerdo, central e direito) são importantes em muitas aplicações. Quando o rato tem apenas dois botões, normalmente «emula-se» um rato com três botões, sendo o botão do meio obtido pela pressão simultânea dos dois botões.

2.1 Entrar na máquina

Para se aceder a uma estação de trabalho Unix é necessário fazer o *login*, ou seja, fornecer à máquina a identificação de um utilizador e uma *password* de autenticação. Este pode ser feito em modo de texto ou em modo gráfico, dependendo da maneira como a máquina está configurada. Na FEUP a grande maioria dos sistemas Unix estão configurados para fazer a validação do utilizador em modo gráfico.

Num *login* gráfico é normalmente apresentado um ecrã com uma imagem de fundo, e uma janela com: um campo para a identificação do utilizador, um campo para a password, e menus variados (que mudam conforme o gestor de *login*) que permitem mudar de gestor de janelas, fechar a máquina, recomeçar, etc.

Para entrar na máquina basta introduzir o seu *username* no campo de identificação do utilizador, a sua *password* no campo para a *password*, escolher o gestor de janelas desejado e carregar em OK.

2.2 Gestores de Janelas

2.2.1 KDE

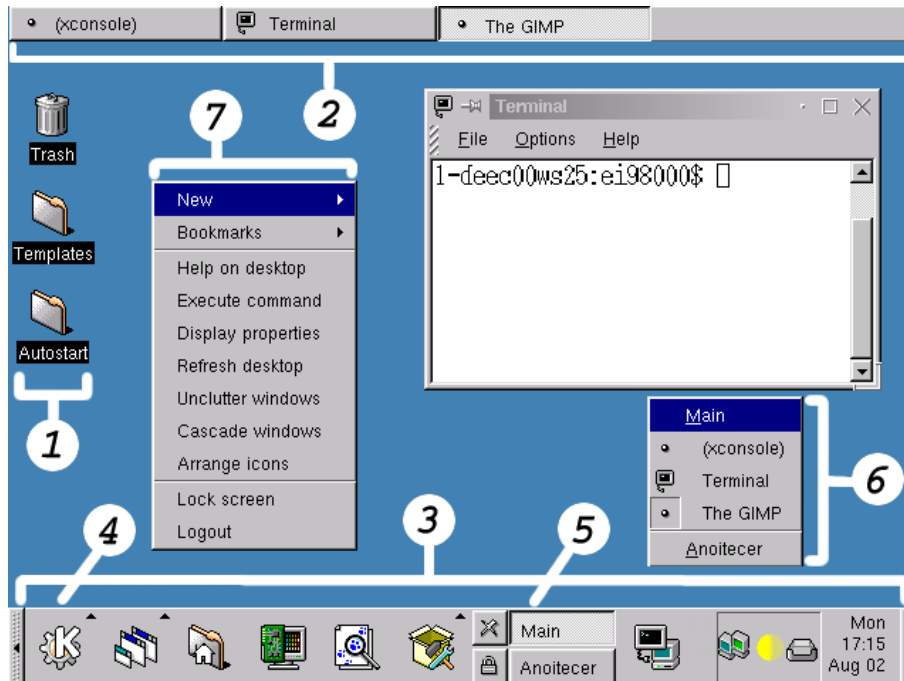



Figura 2.1: Ambiente de trabalho no KDE.

Descrição

KDE é o gestor de janelas mais *user-friendly* disponível em Linux. É o ideal para pessoas habituadas a trabalhar em Windows 9x/NT devido à sua facilidade de utilização e semelhança a nível gráfico. O interface é bastante intuitivo permitindo que pessoas novas ao ambiente trabalhem com eficiência e rapidez.

Interface

Possui um *desktop* onde se podem colocar atalhos (1) para aplicações ou *drives*, uma *taskbar* (2) para comutar facilmente entre as aplicações que estão a correr, e uma barra de ferramentas (3) para executar as principais aplicações mais rapidamente. Todos os programas podem ser corridos através do menu de aplicações (4) acedido no canto inferior esquerdo na barra de ferramentas. Este gestor de janelas tem ainda a vantagem de trazer de raiz uma grande quantidade de aplicações: calculadora, editor de texto, jogos vários, editor de imagens, explorador de fractais, compressor de ficheiros, etc (que podem ser usados também em qualquer outro gestor de janelas diferente do KDE).

Por omissão existem quatro áreas virtuais de trabalho podendo este número ser extendido até oito. A transição entre os vários ambientes virtuais é feita através de botões na barra de aplicações (5). Cada ambiente de trabalho pode ser configurado independentemente com diferentes imagens de fundo, ícones e até nomes.

O botão central e o botão direito do rato permitem-nos aceder a diferentes menus. Pressionando o central temos acesso a um menu (6) com as aplicações a correr nos vários ambientes de trabalho. O botão direito, por sua vez, dá-nos acesso a um menu de contexto (7) com opções específicas relativas à área onde se encontra o cursor do rato nesse momento.

Para sair deste gestor de janelas e voltar ao gestor de *login*, basta executar o comando `Logout` que se encontra no menu de aplicações (4).

Configuração

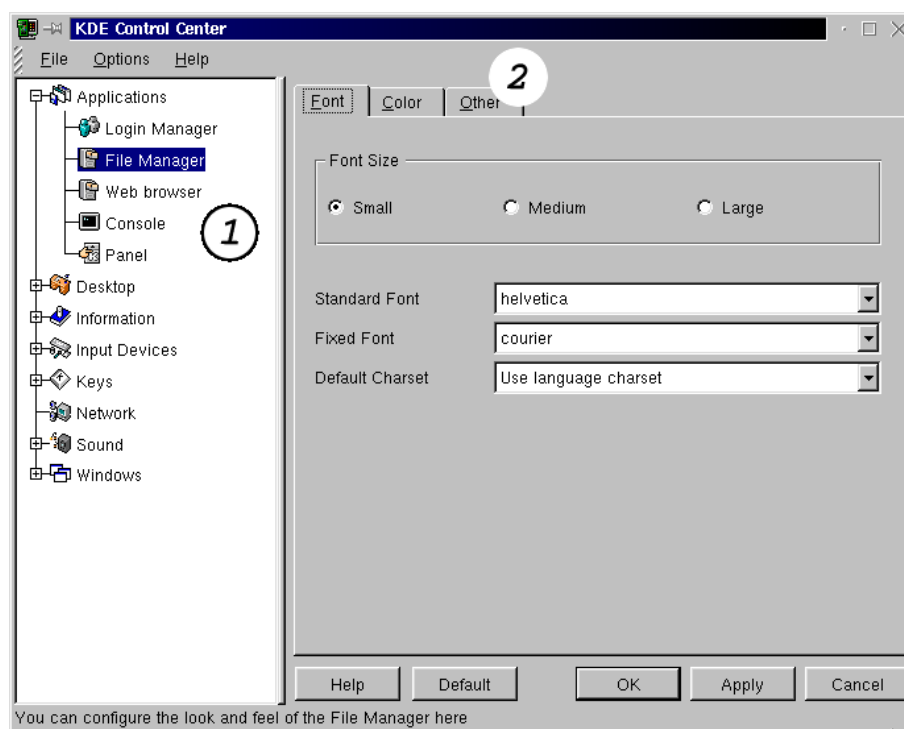



Figura 2.2: KDE Control Center.

A configuração do KDE é muito fácil e totalmente controlada através de uma única aplicação denominada «KDE Control Center». Aqui pode-se alterar todos os aspectos do gestor de janelas como a imagem de fundo, o número de ambientes de trabalho e os seus nomes, a língua utilizada, o gestor de ficheiros, etc. A navegação entre as várias opções é fácil e rápida devido ao seu esquema em árvore (1) e ao uso de separadores (2).

2.2.2 ICEwm

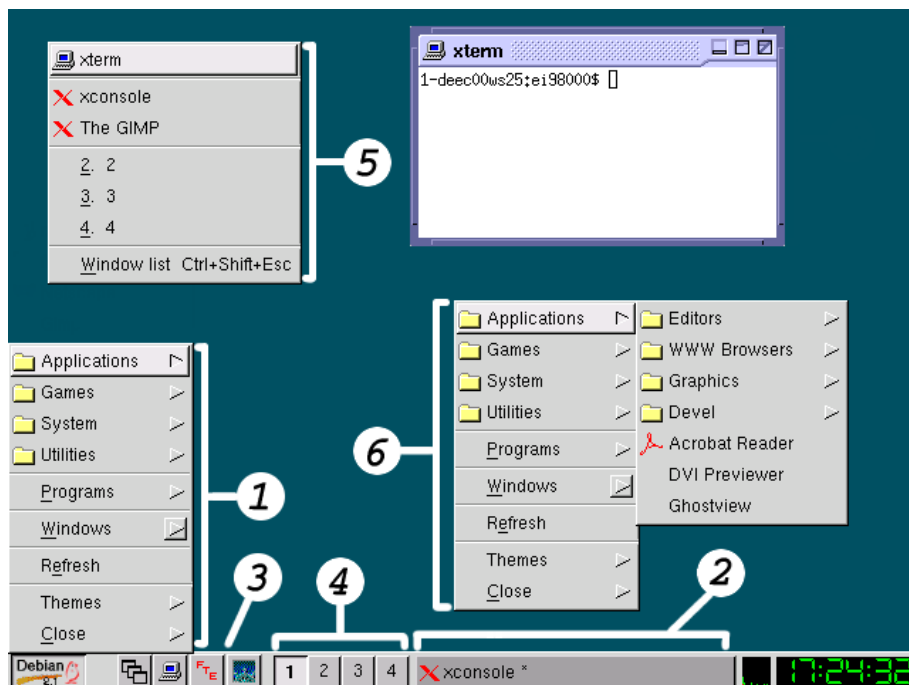



Figura 2.3: Ambiente de trabalho no ICE.

Descrição

O ICE, segundo o seu autor, foi criado com o objectivo de fornecer aos utilizadores de Linux um gestor de janelas que fosse pequeno, rápido e parecido com os interfaces gráficos convencionais. Como tal, é um gestor que não traz nenhuma novidade a nível de utilização. O ambiente de trabalho é simples e parecido com o KDE, não representando um obstáculo para utilizadores não familiarizados com este ambiente.

Interface

A nível de interface possui um menu de aplicações (1) que é acessível no canto inferior esquerdo e uma *taskbar*, no fundo do ecrã, que permite comutar entre os vários programas a ser executados (2) e correr as aplicações mais importantes (3).

Tal como no KDE, por omissão, são disponibilizados quatro ambientes de trabalho, podendo este número ser alterado editando o ficheiro «preferences». A comutação entre os vários ambientes de trabalho é fácil e rápida devido à existência de botões na *taskbar* (4) .

Os vários botões do rato dão-nos acesso a diferentes menus. Pressionando o botão esquerdo temos acesso a um menu com as várias aplicações a correr nesse

momento (5). O botão direito, por sua vez, apresenta diferentes opções conforme o local da área de trabalho: no *desktop* mostra-nos o menu de aplicações (6), e em aplicações minimizadas mostra-nos um menu com várias acções que se podem executar na janela da aplicação (mover, alterar tamanho, maximizar, minimizar, etc).

Através dos menus Close -> Exit Window Manager, acessíveis no menu de aplicações (1), poderá sair deste gestor voltando assim para o gestor de *login*.

Configuração

A configuração é totalmente feita através da edição de ficheiros de texto[1]. Estes devem-se localizar na conta pessoal dentro do subdirectório «.icewm/». Dentro deste deverão existir os ficheiros:

- preferences - opções gerais como *paths*, cores, fontes, número de ambientes de trabalho.
- menu - configuração do menu de aplicações.
- programs - contém um menu gerado automaticamente com as aplicações *standard*.
- winoptions - configuração de aplicações: ícone, a existência de caixilho ou barra de título, etc.

[1] A edição de ficheiros de texto será abordada mais à frente neste capítulo.

2.2.3 Window Maker

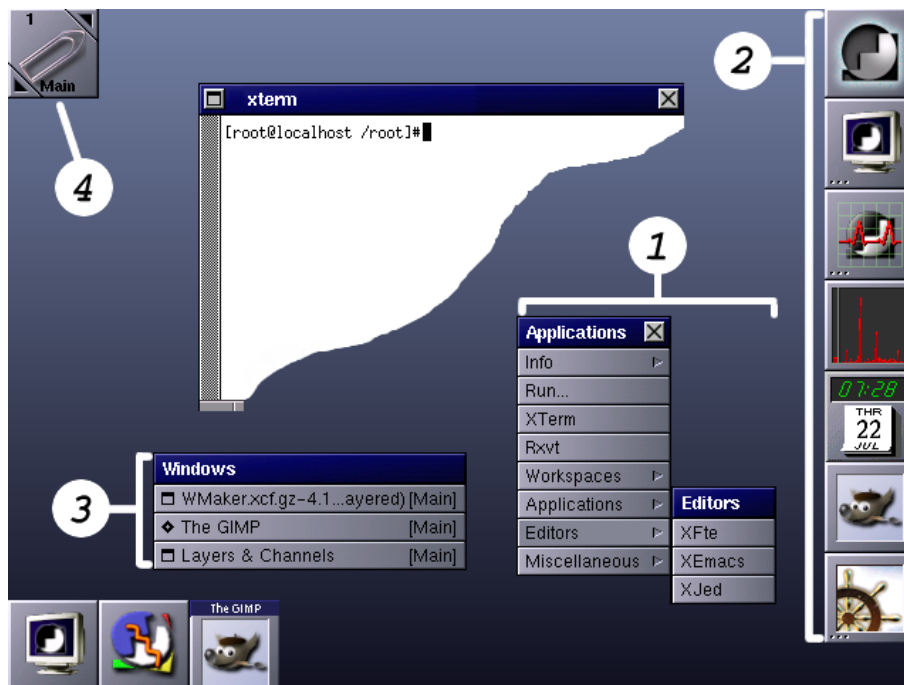



Figura 2.4: Ambiente de trabalho no Window Maker.

Descrição

Este gestor de janelas é totalmente diferente e inovador para quem nunca tenha trabalhado no sistema operativo NextStep, o qual tenta emular. Foi desenhado para ser rápido e leve para o sistema, mas ao mesmo tempo poderoso e agradável graficamente.

Interface

Neste gestor, ao contrário do KDE, o menu de lançamento de aplicações (1) é acessível em qualquer altura bastando para isso pressionar o botão direito do rato em qualquer ponto da área de trabalho. Para tornar a execução das aplicações principais mais fácil existe ainda uma barra denominada *Dock* (2) onde se podem «colar» ícones (que quando pressionados executam as respectivas aplicações) ou aplicações específicas que ficam acopladas a esta barra durante a sua execução (relógios, gráficos da utilização do processador, etc). Pressionando o botão direito do rato sobre um dos ícones dentro do *Dock*, entra-se num menu que permite a sua configuração: opções, propriedades da janela, lançamento automático ao arrancar, etc. O botão central quando pressionado dá acesso a um menu (3) com as aplicações a correr nesse momento nos vários ambientes de trabalho.

Ao contrário de outros gestores de janelas o Window Maker apresenta, por omissão, apenas uma área de trabalho podendo o utilizador acrescentar outras à medida que vai precisando. Para tal é apenas necessário criar uma nova através do respectivo menu (Workspace -> Workspaces -> New). Após ter várias áreas de trabalho definidas a comutação faz-se através do *clip* (4) (normalmente no canto superior esquerdo) ou das teclas de atalho definidas na configuração (normalmente Alt-n, onde n é o número da área virtual).

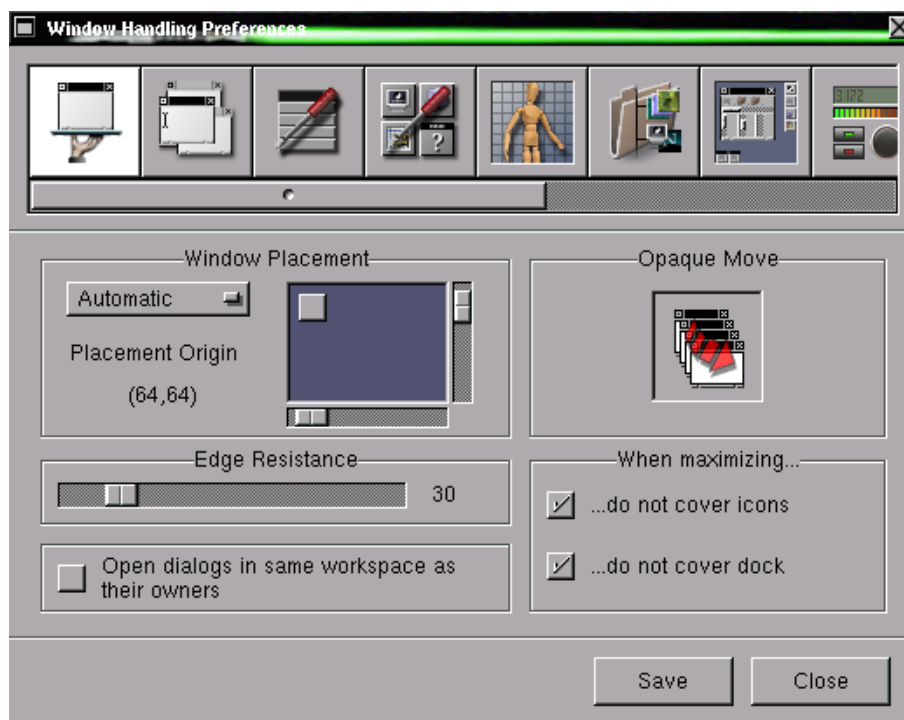
Os menus Window Managers -> Exit, acessíveis no menu de aplicações (1), permitem sair do gestor, tornando acessível o gestor de *login*.

Configuração

Figura 2.5: Programa de configuração do Window Maker.

A configuração deste gestor de janelas, à semelhança com o KDE, está integrada numa só aplicação denominada «Window Maker Preferences». Apesar da configuração poder ser feita alterando directamente ficheiros de texto, esta aplicação torna tudo muito mais fácil implementando quase por completo todas as opções que o Window Maker apresenta.

2.2.4 Fvwm

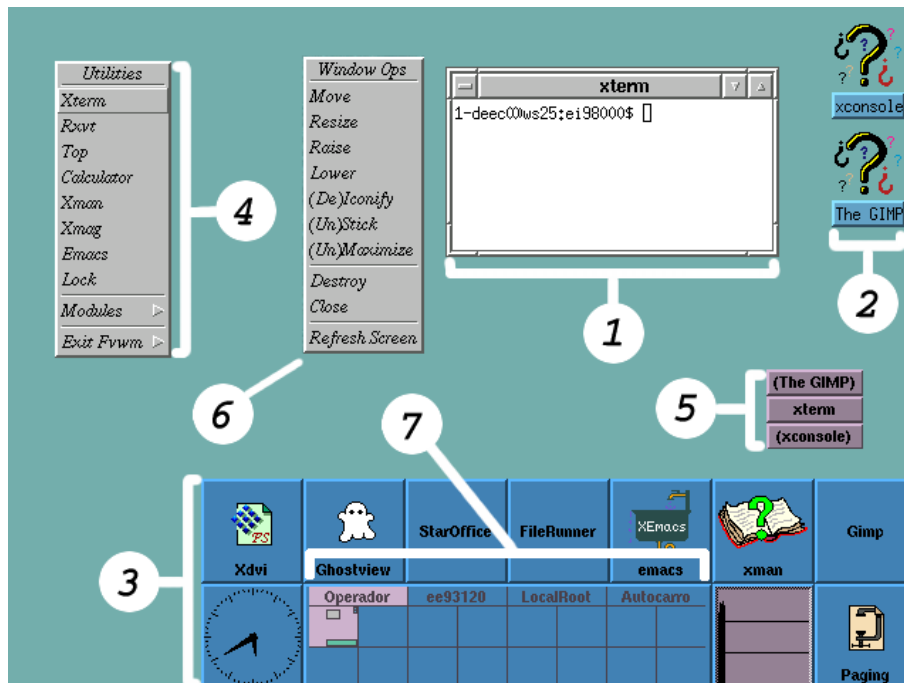



Figura 2.6: Ambiente de trabalho no Fvwm.

Descrição

O fvwm é o gestor de janelas aqui apresentado mais próximo da filosofia Unix. filosofia O seu interface é muito simplista e, como tal, esta totalmente orientado para o trabalho e para a rapidez não representando uma carga para o sistema.

Interface

O interface deste gestor foi pensado para ser modular, ou seja, é construído de pequenos programas muito específicos. Todo o conjunto destes programas forma o interface final. Assim este torna-se totalmente configurável podendo adaptar-se as necessidades e ao gosto do utilizador.

Este gestor apresenta um *desktop* que é utilizado para as janelas das aplicações (1) e para os ícones dos programas que estão a correr minimizados (2). No fundo deste existe ainda um módulo (3) (denominado «Goodstuff») que permite correr as principais aplicações através de botões que lhe estão acoplados.

Premindo o botão esquerdo do rato temos acesso ao menu de aplicações (4) enquanto o botão direito mostra um menu com os programas que estão a ser executados nesse momento (5). O botão central dá acesso a um menu (6) com várias acções que se podem aplicar a uma janela: mover, redimensionar, minimizar, etc.

O número de ambientes virtuais é definido no ficheiro de configuração e a comutação entre eles é feita através de um módulo denominado «Pager» (7) que normalmente está acoplado ao módulo «Goodstuff».

Para sair deste gestor basta executar o comando Exit fvwm -> Yes, Exit fvwm, acessível no menu de aplicações (4), tornando-se assim acessível o gestor de *login*.

Configuração

A sua configuração é totalmente feita através de um ficheiro de texto[2] (.fvwmrc) que permite alterar todo e qualquer aspecto do gestor de janelas: o fundo, as aplicações e módulos a correr no ínicio de cada sessão, a existência de caixilho ou barra de título, a fonte utilizada, o número de ambientes virtuais, etc.

[2] A edição de ficheiros de texto será abordada mais à frente neste capítulo.

2.3 Edição e Visualização de ficheiros

2.3.1 HTML

O HTML (HyperText Markup Language) é o formato mais utilizado no mundo para a publicação de páginas na Internet. É um formato livre baseado no SGML (*Standard Generalized Markup Language*)[3] e pode ser criado por um grande número de ferramentas diferentes, desde simples editores de texto até editores WYSIWYG [4] mais avançados. Utiliza um sistema de comandos (designados de *tags* no html) que faz a formatação das frases e caracteres, atribui links, etc. Em Linux são inúmeras as aplicações disponíveis para visualizar ficheiros deste tipo, sendo apresentadas algumas a seguir.

Netscape Communicator

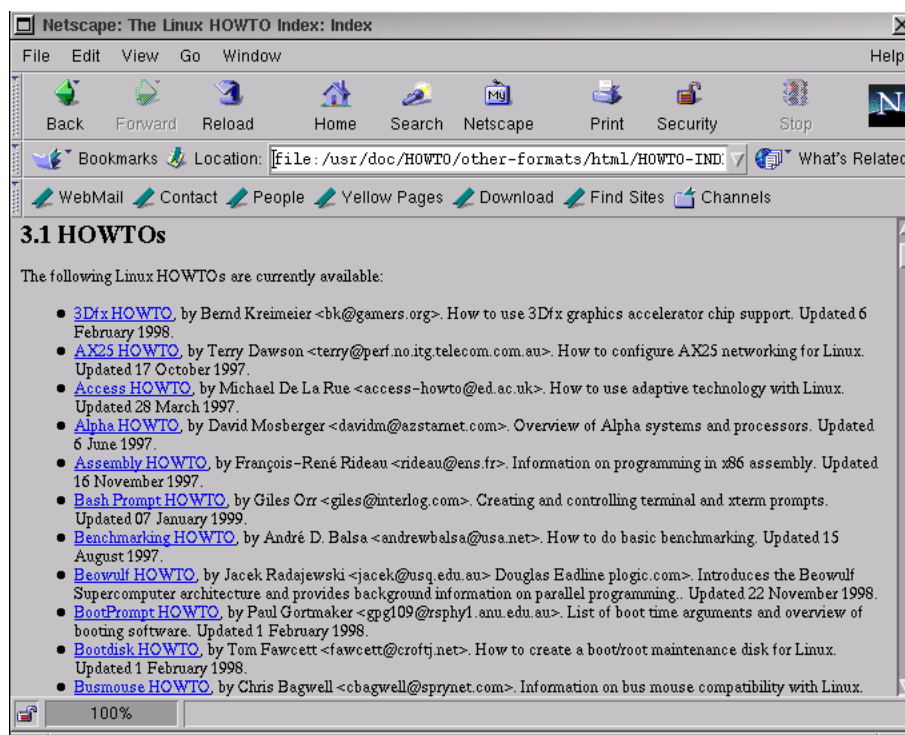



Figura 2.7: Netscape Communicator.

Este é o browser mais utilizado devido a estar presente na maioria das distribuições e de ser o mais completo disponível em Linux. Tem um ambiente gráfico bastante intuitivo e profissional e é compatível com as últimas versões de HTML e Java.

[3]Uma «meta-linguagem» utilizada em projectos de documentação a grande escala.
[4]What You See Is What You Get

Para abrir uma página em `Netscape`, basta executar os seguintes passos:

1. Abrir o Netscape através dos menus do gestor de janelas Personal -> Netscape Communicator.

2. Abrir o menu File -> Open Page.

3. Escrever o endereço (internet ou local) da página HTML na caixa de diálogo.

Lynx

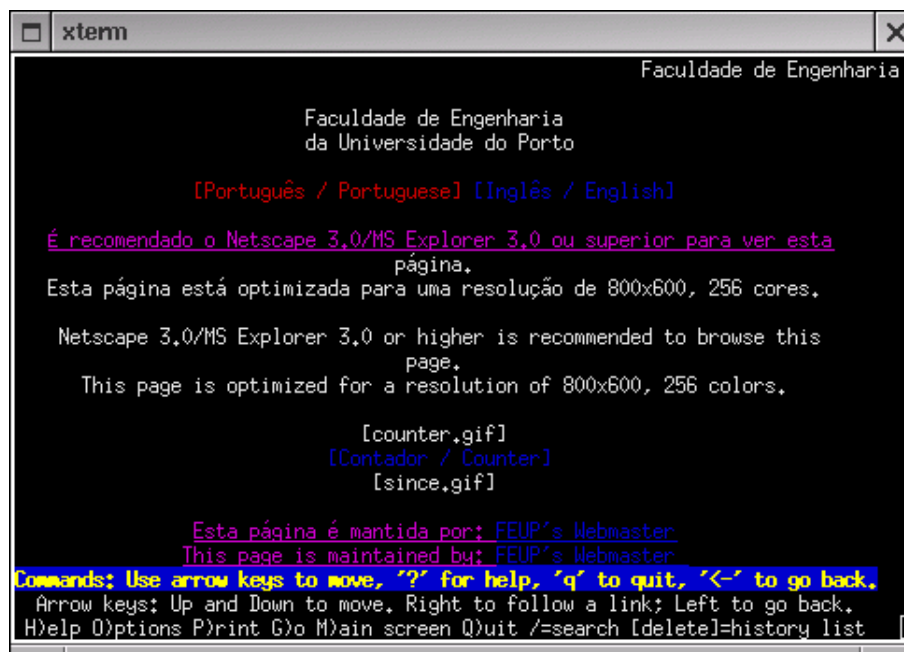



Figura 2.8: Lynx.

O `Lynx` é um browser extremamente modesto mas ao mesmo tempo eficiente. Trabalha apenas em modo de texto, o que o impede de ler imagens, JavaScript, etc. Em vez disso, apresenta uma referência para os elementos que não consegue ler. Devido ao facto de trabalhar em modo de texto, é extremamente rápido e valioso em situações de sobrecarga do processador ou da rede. No fundo do ecrã temos acesso a uma lista das várias opções disponíveis, bastando pressionar a tecla destacada para executar uma dada acção.

Para abrir uma página em `Lynx`, seguir os seguintes passos:

1. Abrir o `Lynx` através dos menus Personal -> Lynx.

2. Pressionar «g» e escrever o endereço da página desejada.

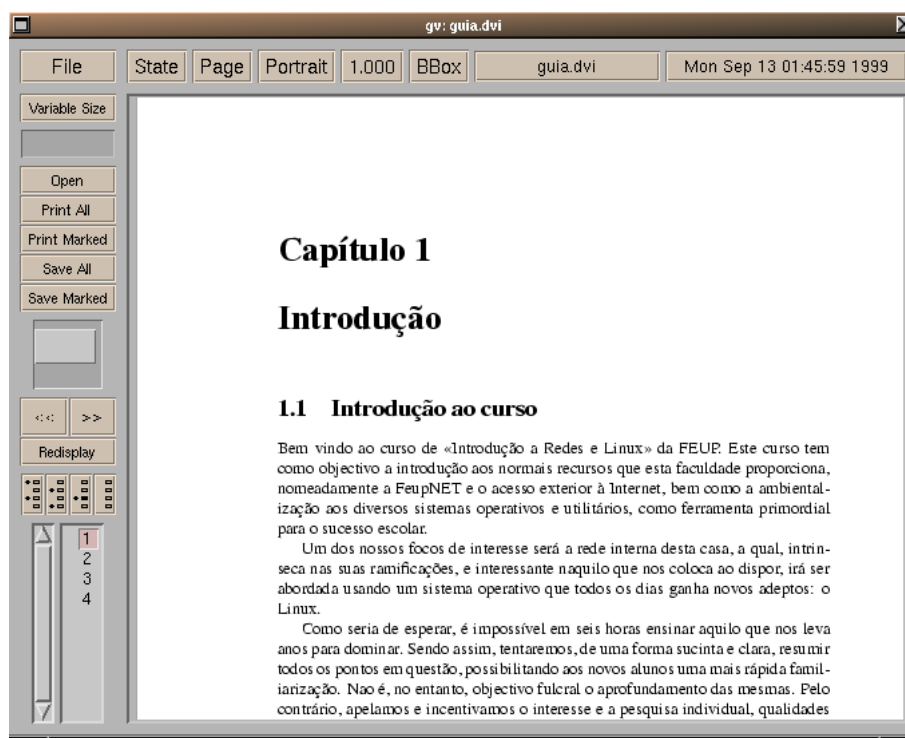



Figura 2.9: GV.

2.3.2 PostScript

O PostScript é uma linguagem de programação, criada em 1985 pela Adobe, e optimizada para imprimir gráficos e texto, ou seja, é uma linguagem de descrição de página. O seu principal objectivo é fornecer uma linguagem conveniente para descrever imagens de um modo independente da máquina utilizada, ou seja, não recorrendo a características específicas de uma dada arquitectura. Na linguagem PostScript, as imagens são construídas em forma vectorial, definindo geometricamente cada curva, em contraste com os gráficos *bitmap* nos quais a figura é obtida pela definição de cada *pixel* num quadriculado da área total. No entanto, um ficheiro PostScript pode também incluir uma imagem *bitmap* perdendo assim a sua propriedade de ser independente da resolução.

Para visualizar o resultado de um ficheiro em PostScript, é normalmente utilizado o programa `gv`. Esta aplicação, derivada do GhostView, tem um interface muito simples, e apresenta apenas o essencial.

Para abrir um ficheiro PostScript com o `gv`, executar os seguintes passos:

1. Abrir o `gv` através dos menus Personal -> GV.

2. Abrir o ficheiro através do botão Open, ou através do menu File -> Open, e seleccionar o ficheiro a ser visualizado.

2.3.3 PDF

O PDF (Portable Document Format) é um formato criado pela Adobe baseado no PostScript. Foi estruturado para ser independente da plataforma, facilmente impresso, visualizável e navegável. A sua visualização é fiel ao documento impresso tornando-se quase uma fotografia do original. É um formato bastante divulgado na Internet por ser um formato livre, o que faz com que existam inúmeras plataformas que o suportam, e por existirem *plug-ins* para os principais browsers que permitem a visualização de ficheiros PDF.

O PDF é uma boa opção quando as limitações do HTML impedem a criação de um determinado documento. Por exemplo, as equações matemáticas não podem ser escritas em HTML; uma solução seria criar um gráfico para cada equação, mas isso torna a página bastante lenta de carregar e produz um resultado de baixa qualidade.[5]

As aplicações mais utilizadas para ver PDFs em Linux são o `Adobe Acrobat Reader` e o `Xpdf`.

Adobe Acrobat Reader

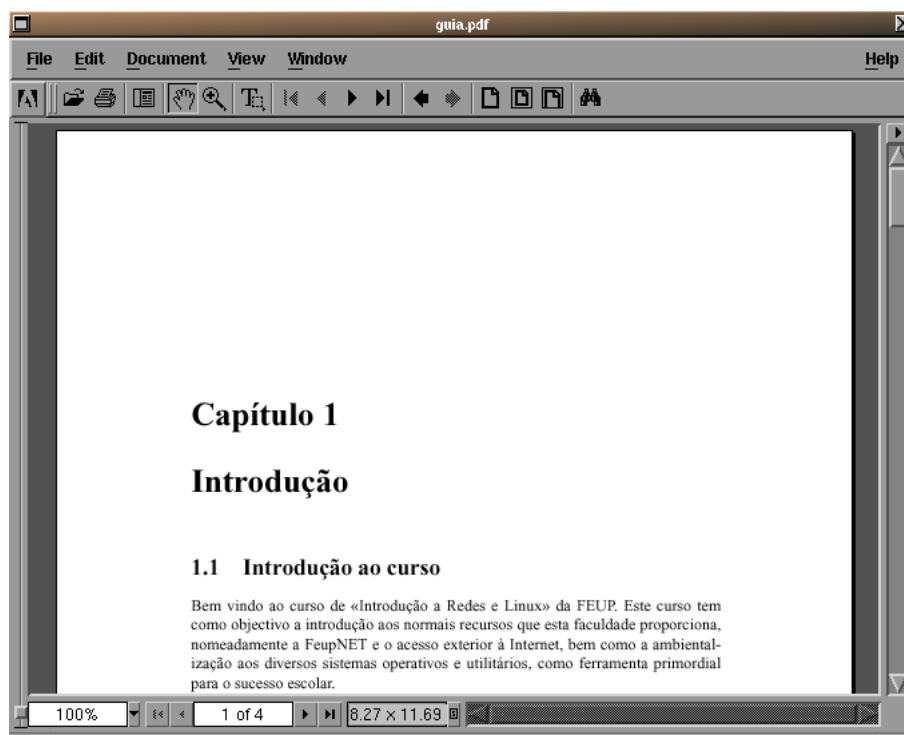



Figura 2.10: Adobe Acrobat Reader.

O `Adobe Acrobat Reader` é o mais completo programa para ver PDFs,

[5] Apesar de o HTML 4 ter suporte para equações matemáticas, muitos dos *browsers* mais conhecidos e actuais ainda não aceitam estes comandos.

já que este formato foi criado pela Adobe. O interface é intuitivo e permite uma navegação simples e rápida dentro do documento a ser visualizado.

Para abrir um ficheiro PDF no `Acrobat Reader` seguir os seguintes passos:

1. Correr o programa através dos menus do gestor de janelas Personal -> Acrobat Reader.

2. Abrir o documento através do menu File -> Open.

Xpdf

Capítulo 1

Introdução

1.1 Introdução ao curso

Bem vindo ao curso de «Introdução a Redes e Linux» da FEUP. Este curso tem como objectivo a introdução aos normais recursos que esta faculdade proporciona, nomeadamente a FeupNET e o acesso exterior à Internet, bem como a ambientalização aos diversos sistemas operativos e utilitários, como ferramenta primordial para o sucesso escolar.

Um dos nossos focos de interesse será a rede interna desta casa, a qual, intrinseca nas suas ramificações, e interessante naquilo que nos coloca ao dispor, irá ser abordada usando um sistema operativo que todos os dias ganha novos adeptos: o Linux.

Figura 2.11: Xpdf.

O `Xpdf` é um visualizador de PDF mais modesto. Apresenta apenas o necessário (mudar de página, *zoom*, imprimir) através de um interface gráfico simples e intuitivo. Os menus são acedidos premindo o botão direito do rato muito ao estilo de gestores de janelas tipo Window Maker e AfterStep.

Para abrir um ficheiro PDF no `Xpdf`, executar os seguintes passos:

1. Correr o programa através do menu Personal -> Xpdf.

2. Abrir o documento utilizando o comando Open, que pode ser acedido premindo o botão direito do rato na janela do Xpdf.

2.3.4 DVI

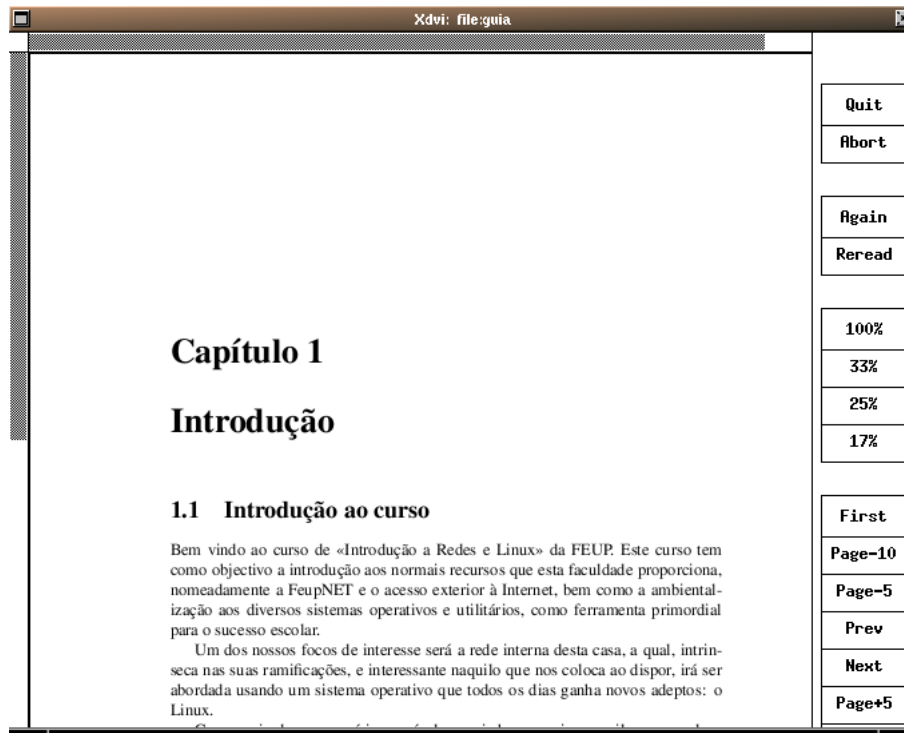



Figura 2.12: Xdvi.

O DVI (DeVice Independent) é um formato que normalmente resulta da compilação de ficheiros na linguagem de descrição de página TeX ou LaTeX, e que, apesar do DVI ser bastante portável, é normalmente convertido para PostScript ou PDF.

A aplicação mais utilizada para visualizar um ficheiro DVI e que se encontra na grande maioria das distribuições é o `xdvi`.

Para visualizar um ficheiro DVI utilizando o `xdvi`:

1. Executar a aplicação através dos menus do gestor de janelas Personal -> Xdvi.
2. Seleccionar o ficheiro desejado na caixa de diálogo.

2.3.5 Ficheiros de texto

Os ficheiros com nome a terminar em `.txt` costumam ser ficheiros de texto ASCII não formatado, ou seja, um formato completamente universal e independente da máquina, e que é utilizado para diversos fins como documentação ou configuração de programas.

Há vários métodos e aplicações para visualizar ficheiros de texto mas apresentaremos aqui uma aplicação que aproveita o ambiente gráfico em que está inserida:

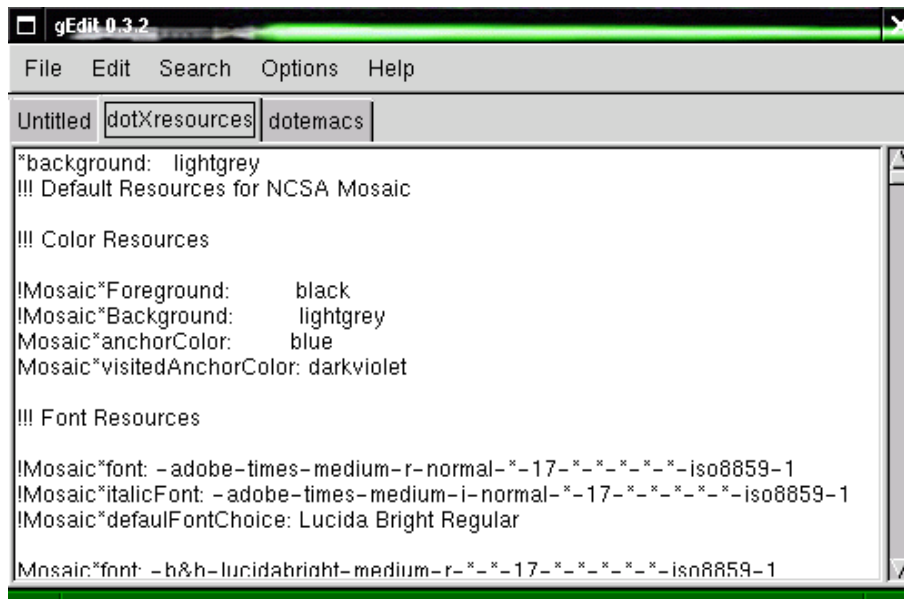



Figura 2.13: gEdit.

o `gEdit`. Este programa apresenta um interface simples e agradável, e permite a visualização de vários ficheiros através de um práctico sistema de separadores. Apresenta ainda a possibilidade de busca dentro do ficheiro a ser visualizado e de este ser alterado, tornando-se uma ferramenta práctica e indicada para pequenas alterações em ficheiros deste tipo.

Para visualizar ou editar um ficheiro de texto com o `gEdit`:

1. Executar a aplicação através do menu Personal -> gEdit.
2. Abrir o menu File -> Open.
3. Selecionar o ficheiro desejado na caixa de diálogo.

2.3.6 Texto formatado

Tal como em outros sistemas operativos também o Linux tem à sua disposição ferramentas de edição de documentos. Um dos mais divulgados em Linux é o `StarOffice`. Este não é apenas um programa de edição de texto mas sim um conjunto de ferramentas de escritório totalmente integradas numa só aplicação e que permitem editar quase todo o tipo de documentos: folha de cálculo, base de dados, imagem, HTML, diagramas, fórmulas, apresentações, correio electrónico, etc. Este programa pretende não ser apenas mais uma ferramenta mas sim uma área de trabalho completa. Como tal apresenta um *desktop* onde se podem colocar atalhos para os documentos mais utilizados e um menu acessível no canto inferior esquerdo, muito ao estilo do Windows, onde se podem correr os vários programas ou abrir documentos pessoais.

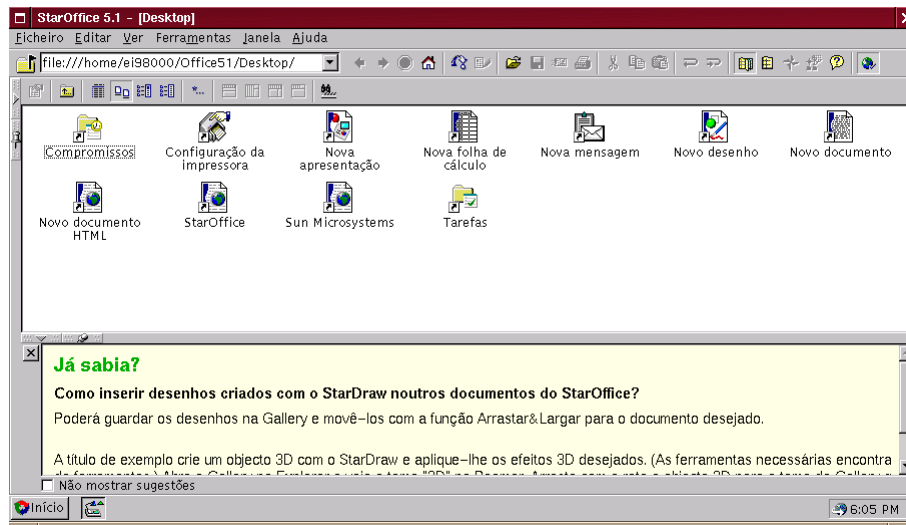



Figura 2.14: StarOffice.

O StarOffice tem total compatibilidade com os pacotes de ferramentas de escritório mais usuais tais como o Microsoft Office e Lotus SmartSuite, podendo haver assim uma maior flexibilidade e interacção entre utilizadores de ferramentas e sistemas operativos diferentes.

Para começar a trabalhar no `StarOffice`:

1. Correr o `StarOffice` através do menu do gestor de janelas Personal -> StarOffice.

2. Carregar no ícone que corresponde ao tipo de documento que quer criar ou

3. Correr o programa correspondente ao tipo de documento que quer criar através do menu no canto inferior esquerdo.

2.3.7 Imagens

O Linux suporta todos os tipos de imagens mais usuais. Existem inúmeros programas disponíveis para a visualização de imagens, mas os mais divulgados e existentes em inúmeras distribuições são o `xv` e o `Image Magick`. Estes além de permitirem a visualização permitem ainda converter entre vários formatos, tirar *screenshots*, aplicar filtros, imprimir, etc.

A edição de imagens normalmente é feita com o `Gimp`, um programa de tratamento digital de imagem que permite atingir níveis quase profissionais muito próximos daqueles atingidos com o Adobe Photoshop ou o PhotoPaint da Corel.

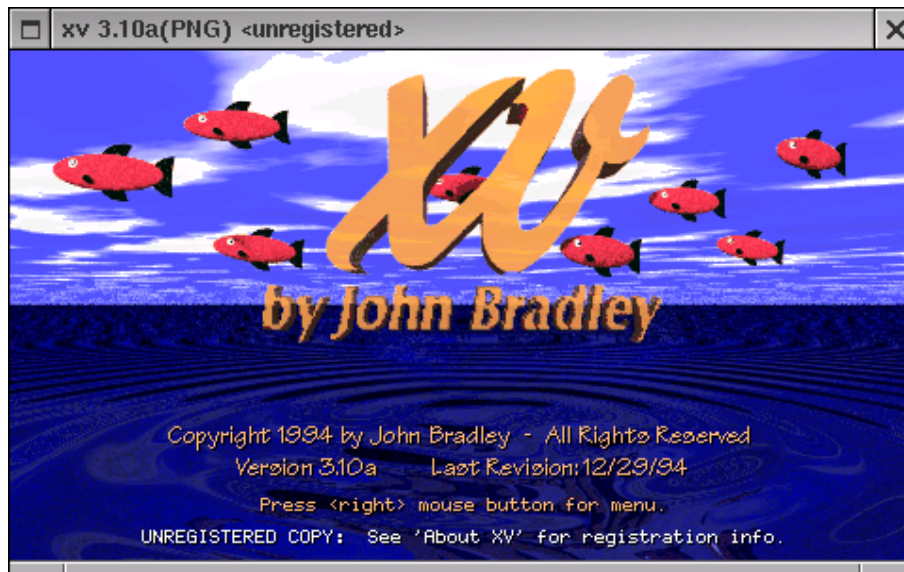



Figura 2.15: XV.

XV

O `xv` é um programa que permite a ver imagens de uma forma simples e rápida. Suporta todos os formatos mais usuais e permite além da ver a imagem, imprimir, aplicar efeitos como blur e sharpen, tirar *screenshots*, etc.

Para visualizar uma imagem em `xv`:

1. Correr o programa através do menu Personal -> XV.
2. Pressionar o botão direito do rato e seleccionar *Load* na caixa de diálogo.

Image Magick

O `Image Magick` é mais do que um visualizador. É um conjunto de aplicações de consola que permitem a manipulação de imagens. Com este pacote pode-se visualizar, converter, capturar, fundir, aplicar efeitos, criar *thumbnails* e imprimir.

Para visualizar uma imagem com o `Image Magick`:

1. Correr o visualizador de imagens do Image Magick através dos menus do gestor de janelas Personal -> Image Magick.
2. Seleccionar o ficheiro na caixa de diálogo.

Gimp

O `Gimp` é o melhor programa de edição de imagem disponível para Linux. Tem todas as potencialidades de um programa de tratamento de imagem profissional suportando *layers*, filtros vários, *plug-ins*, e *scripts* em scheme que se podem aplicar

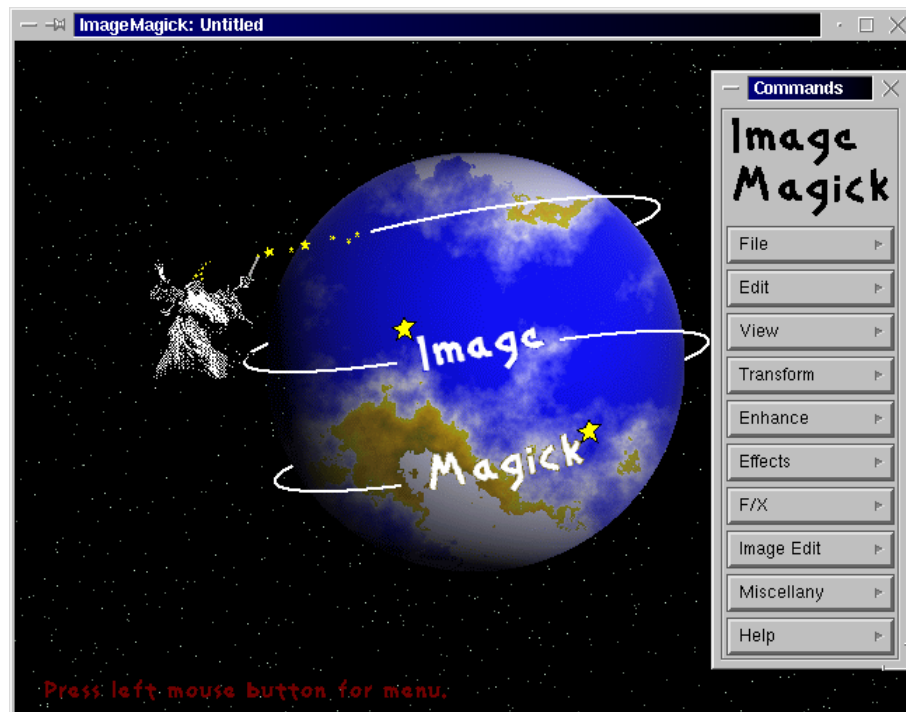



Figura 2.16: Image Magick.

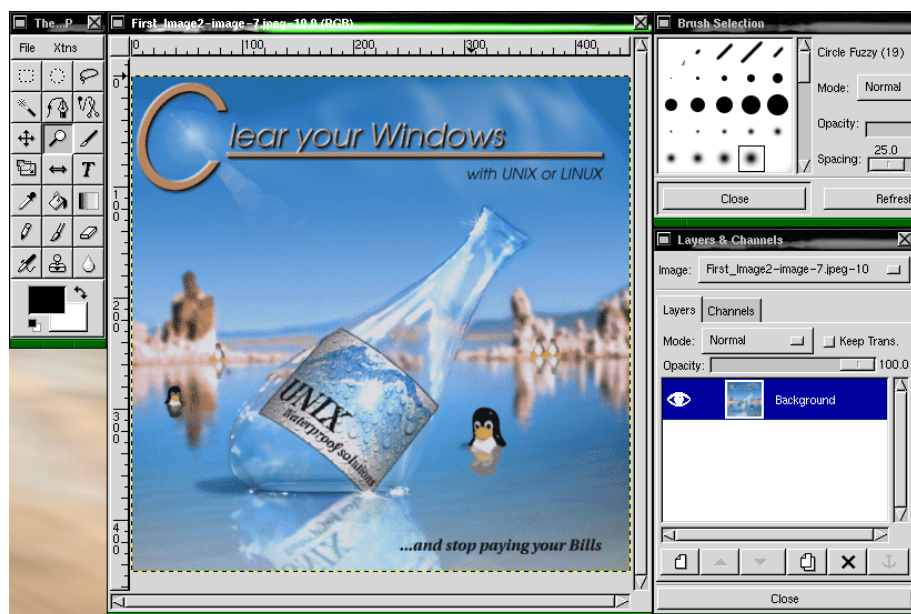



Figura 2.17: Gimp.

às imagens utilizando funções internas do `Gimp`. Estas funções estão organizadas numa base de dados (DB Browser) que apresenta uma pequena descrição de cada função, os respectivos argumentos de entrada e o seu tipo. O Gimp apresenta ainda inúmeras ferramentas de desenho, selecção e retoque de imagem. Suporta os formatos mais usuais, assim como o seu próprio formato (.xcf), e os formatos de compressão `gzip` e `bzip2` para reduzir o tamanho das imagens ao mesmo tempo que grava ou lê.

Um exemplo da utilização do `Gimp` são os *screenshots* desta sebenta. Foram totalmente produzidos com este programa, tanto na captura dos ecrãs como na alteração dos mesmos.

Para produzir as imagens finais dos gestores de janelas utilizaram-se *layers*, para separar os vários elementos da imagem: imagem de fundo, os elementos do interface do gestor de janelas, os balões, e os números. Isto permitiu que a alteração de cada um destes elementos fosse fácil e rápida, pois estando em *layers* separadas, quando dois elementos se sobrepõem, a informação do elemento que fica em baixo não se perde. Assim, quando era necessário mover um menu ou a janela de uma aplicação, a imagem de fundo que estava por baixo permanecia intacta. Uma outra facilidade que as *layers* e o `Gimp` permitem é a selecção rápida dos vários elementos através do comando Select -> Float. Este comando transforma uma selecção num elemento flutuante que se pode mover dentro da imagem. O truque está no facto de cada *layer* ser transparente nas zonas em que não tem informação de elementos, setas ou números. Ou seja, se desejarmos seleccionar o número cinco, por exemplo, podemos fazer uma selecção totalmente arbitrária à volta do número (e que não englobe outros números) e aplicar o comando Select -> Float. Como dentro da nossa selecção só há o número cinco e transparência, ou seja, o número e ausência de informação, este comando pode saber a forma exacta do número e transformá-lo num elemento flutuante. Tudo sem um grande esforço da nossa parte para obter uma selecção perfeita.

O `Gimp` tem muitas mais ferramentas e opções que permitem a fácil manipulação e selecção de imagens, mas o aprofundamento destas técnicas está fora do âmbito do nosso curso. Se estiver interessado em manipulação e tratamento digital de imagem com o `Gimp`, pode obter muita informação na Internet e ler o manual completo totalmente gratuito.

Para editar uma imagem com o `Gimp`:

1. Correr o `Gimp` através do menu Personal -> Gimp.

2. Criar uma nova imagem através do menu File -> New ou

3. Abrir uma imagem através do menu File -> Open.

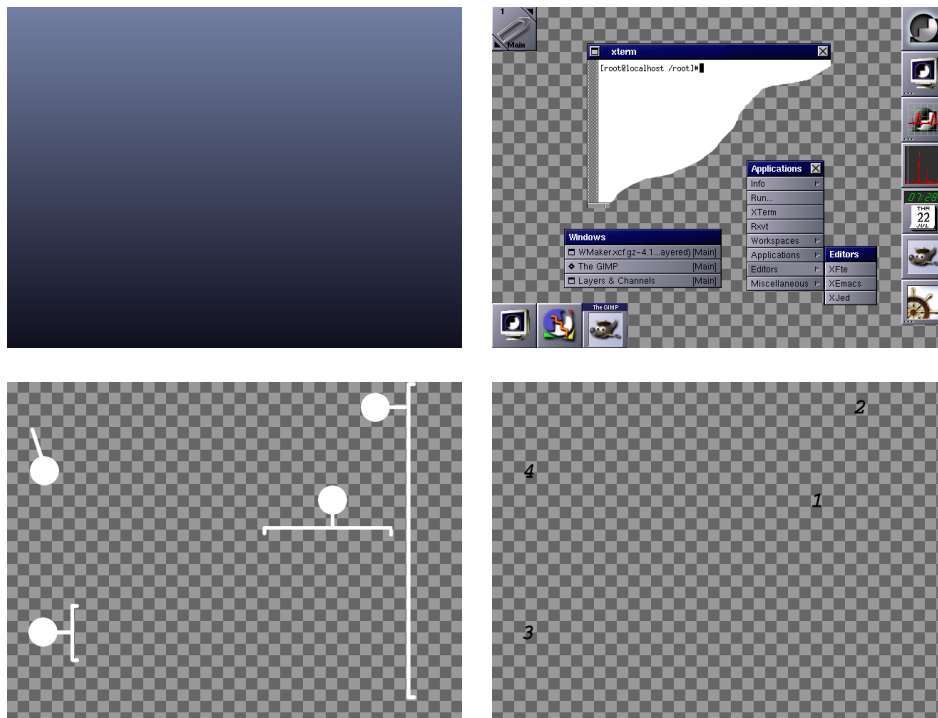



Figura 2.18: As várias *layers* que formam o *screenshot* do Window Maker.

Capítulo 3

Comandos Unix

Quando o Unix foi desenvolvido na década de 1970, a maior parte dos terminais usados eram terminais de modo texto que não permitiam trabalhar no modo gráfico. A execução de programas era feita através da interacção com uma **shell**, que é a interface usada para interpretar comandos introduzidos através do teclado, executar funções dentro do *kernel* do sistema operativo e enviar resultados para o terminal.

A *shell* continua a ser de grande utilidade nos terminais gráficos. O modo gráfico simplifica a interacção com uma aplicação por meio de menus que evitam que o utilizador tenha que lembrar nomes de comandos; mas em contrapartida o utilizador fica limitado a usar o reduzido número de comandos que apareçam nos menus, e não os pode combinar nas diversas formas que são possíveis para quem trabalha numa shell de Unix. Uma shell do Unix permite também a criação de **scripts**, que podem ser desde simples ficheiros com sequências de comandos predefinidos, até complicados programas com execução condicionada de diversos comandos.

3.1 Consolas ou terminais

Uma shell precisa da existência de uma «consola» de modo texto que serve de interface entre o utilizador e a *shell*. Normalmente em Linux existem já 6 consolas virtuais predefinidas (designados de `tty1` até `tty6`) que podem ser acedidas com as teclas Alt-F1, até Alt-F6. Quando se está a trabalhar em X (no terminal virtual número 7), é preciso carregar simultaneamente as teclas Ctrl+Alt+Fn, onde n é o número da consola virtual; para regressar ao X, usa-se Alt-F7. Cada consola virtual apresenta uma mensagem de `login`, que permite entrar no sistema e abrir uma shell (a shell usada costuma ser `bash`, mas existem várias versões da shell). Várias shells podem executar em simultâneo, independentemente, e até pertencendo a diferentes utilizadores.

Uma outra forma de abrir uma shell quando se está a trabalhar em X, consiste em usar opções de menu que abre uma consola com uma sessão duma shell com o mesmo nome do utilizador de quem está a executar o programa X. Dentro dos

menus do gestor de janelas aparecem algumas consolas como por exemplo o Xterm, o Rxvt e o Eterm.

3.2 Alguns comandos básicos

A primeira coisa que devemos explicar é como é que o sistema de ficheiros esta estruturado.

3.2.1 O sistema de ficheiros

O sistema de ficheiros (*filesystem*) é basicamente um sistema organizado e hierárquico no qual são guardados dados sobre a forma de directórios e ficheiros. Existem vários tipos, mas o que iremos estudar é o Ext2FS (*Extended 2 Filesystem*) usado com maior frequência nos sistemas Linux.

Uma analogia muito elucidativa é a de que um directório pode ser comparado a uma gaveta. Dentro destas podem existir ficheiros ou ate mesmo outras gavetas, as quais por sua vez possuem as mesmas propriedades. Cada um destes objectos tem um nome associado, normalmente indicativo da sua função. Estes directórios e ficheiros estão todos como que «guardados» numa grande gaveta, a qual chamamos de directório raiz. Esta forma de organização forma o conceito de hierarquia.

Usa-se uma barra diagonal / para separar os nomes de directórios e subdirectórios. A distribuição e nomes dos directórios segue alguns padrões gerais com algumas variações. Um sistema típico de ficheiros de Linux pode apresentar a seguinte organização:

```
MAPA                         Representação Textual
 /                               /
 |- boot                         /boot
     |- vmlinuz                  /boot/vmlinuz
     |- System.map               /boot/System.map
 |- dev                          /dev
     |- hda1                     /dev/hda1
     |- fd0                      /dev/fd0
     |- ttyS0                    /dev/ttyS0
 |- etc                          /etc
     |- conf.modules             /etc/conf.modules
     |- init.d                   /etc/init.d
     |- lpd                      /etc/init.d/lpd
 |- home                         /home
     |- ei92001                  /home/ei92001
```

O directório / é o directório raiz. O directório */etc* é onde são armazenados os ficheiros de configuração do sistema e unicamente o superusuário pode alterá-los. O directório */dev* é especial no sentido de que os seus ficheiros são realmente dispositivos do sistema; assim, por exemplo, se enviarmos dados para o ficheiro */etc/fd0* estaremos realmente enviando dados para o disco flexível. No directório */home* aparecem os subdirectórios dos utilizadores, onde estes têm permissão para criar e alterar ficheiros. Uma forma abreviada de representar */home/ei92001* (directório de trabalho do utilizador *ei92001*) é *~ei92001*.

Alguns comandos da *shell* que nos permitem navegar por entre o sistema de ficheiros são os que se seguem:

ls Permite obter uma listagem dos ficheiros e subdirectórios existentes num directório. Por exemplo, `ls` */etc* mostrará todos os ficheiros e subdirectórios do directório */etc*. Sem nenhum argumento, `ls` mostra o conteúdo do directório corrente. O comando `ls` já pode ter sido configurado com algumas opções como, por exemplo, a opção de representar subdirectórios e diversos tipos de ficheiro em diferentes cores. Uma outra opção muito útil é a opção `-l` (as opções de um comando geralmente são precedidas por um hífen) que indica que será usado um formato mais comprido com mais informação; assim por exemplo, o comando

```
ls -l
```

pode produzir uma listagem como a seguinte

```
total 123
-rw-r--r--  1 root  root    126 Jul 16  1998 XDvi
drwxr-xr-x  2 root  root   5120 Sep 19 17:31 dvips
-rw-r--r--  1 root  root   2165 Jun  3 14:52 fmtutil.cnf
-rw-r--r--  1 root  root   2391 Jun 28 18:39 language.dat
-rw-r--r--  1 root  root    635 Feb  9  1999 mktex.cnf
-rw-r--r--  1 root  root  89910 Nov  2  1998 modes.mf
drwxr-xr-x  2 root  root   1024 Sep 19 17:41 pdftex
-rw-r--r--  1 root  root  18895 Sep 19 17:42 texmf.cnf
```

agora é possível distinguir entre ficheiros e subdirectórios: nos subdirectórios aparece um d na primeira coluna; neste caso existem dois subdirectórios *dvips* e *pdftex*. As letras a seguir à primeira coluna indicam as permissões dos ficheiros e subdirectórios, como veremos mais para a frente. A seguir às letras vem um número que indica quantos outros ficheiros têm sido «enlaçados» ao ficheiro ou subdirectório (número de enlaces simbólicos ou ficheiros dentro de um subdirectório). As duas colunas onde aparece a palavra `root` indicam o nome do dono do ficheiro e o grupo ao qual pertence (neste caso o superusuário ou `root`). A seguir vem um número que indica o tamanho

em bytes; e o total de kilobytes do directório, 123, aparece na primeira linha. Finalmente temos a data de criação ou última modificação, e o nome. É importante salientar que nos nomes distingue-se entre maiúsculas e minúsculas.

Uma outra opção útil é `-A` para mostrar também ficheiros e subdirectórios ocultos (aqueles com nome a começar por ponto).

cd Para mudar o directório corrente, usa-se o comando `cd` (*change directory*). Se acabarmos de obter a listagem obtida acima com `ls -l`, imediatamente a seguir poderíamos deslocar-nos para o directório *dvips* por meio do comando

```
cd dvips
```

Para subir um directório, até o «pai» do directório corrente, usa-se:

```
cd ..
```

Ou podemos simplesmente deslocar-nos a um directório qualquer indicando o seu nome completo:

```
cd /home/ei92001/trabalhos/Janeiro
```

pwd (*Print Working directory*.) Chegará sempre um momento no qual não lembramos o directório corrente onde nos encontramos (embora a *shell* possa estar configurada para mostrar o directório corrente antes do *prompt*). Nesses casos, o comando `pwd` é útil para mostrar o nome completo do directório corrente.

mkdir (*Make Directory*.) Este é o comando usado para criar um subdirectório do directório corrente. Por exemplo:

```
mkdir programas
```

cria um subdirectório com nome *programas*. Embora a sintaxe dos nomes dos ficheiros e subdirectórios seja bastante flexível, convém não exagerar, para não terminar com nomes complicados ou com caracteres especiais que possam confundir à *shell*.

rmdir (*Remove Directory*.) Usado para apagar subdirectórios. Para que funcione, será necessário que o subdirectório a ser apagado esteja vazio. Se o subdirectório não estiver vazio, podemos forçar a sua eliminação por meio do comando:

```
rm -r subdirectório
```

Mas use este comando com discrição já que, como norma geral, ficheiro ou directório que seja apagado será perdido para sempre.

Para poder criar ou apagar subdirectórios será preciso ter permissão para modificar o seu conteúdo. Isto nos traz de volta à informação que aparecia nas primeiras colunas da listagem obtida com `ls -l`. Como vimos, cada ficheiro tem um criador, e um grupo de utilização. Estes são-nos indicados pelas colunas 3 e 4 do exemplo acima. Cada ficheiro ou subdirectório pode ter permissões de leitura (r), modificação (w), incluindo a possibilidade de ser apagado, e de execução (x), definidas independentemente para o dono do ficheiro (u), o grupo de utilização (g), e outras pessoas (o).

Por exemplo, o ficheiro *language.dat* que aparecia na listagem acima tinha a seguinte informação:

```
-rw-r--r-- 1 root root 2391 Jun 28 18:39 language.dat
```

Os três caracteres a seguir à primeira coluna, `rw-` correspondem às permissões do dono (neste caso root), que permitem leitura e modificação mas não execução. Os três caracteres seguintes, `r--`, indicam as permissões para o grupo (um grupo também chamado root neste caso) que unicamente permitem leitura. Finalmente, outros utilizadores têm as mesmas permissões que para o grupo de utilização (`r--`). O superusuário (*root*) normalmente é o único a quem é permitido ultrapassar as suas permissões.

chmod, chown e chgrp Estes três comandos são muito úteis para modificar as propriedades de um ficheiro ou subdirectório. O comando

```
chown nome.grupo ficheiro
```

modifica o «nome» do dono do ficheiro e do «grupo» de utilização de um «ficheiro»; se não for dado um grupo, o grupo continua igual. Quem usa este comando deverá possuir autorização para modificar o ficheiro, ou ser o dono dele, e deverá ter cuidado para não terminar com um ficheiro que ele próprio já não possa alterar por não lhe pertencer. O comando `chgrp` é semelhante mas modifica apenas o grupo e, portanto, precisa apenas do nome do grupo e do ficheiro.

O comando `chmod` usa-se para modificar as permissões de um ficheiro ou directório da seguinte forma: primeiro dizemos a quem vamos modificar as permissões: ‘u’, ‘g’ ou ‘o’. A seguir, indicamos a propriedade que queremos modificar: ’r’, ’w’, ’x’, a seguir usamos ou ‘=’ para definir as permissões exactas, ou ‘-’ ou ‘+’para desactivar ou activar alguma permissão: ‘r’, ‘w’ ou ‘x’. Uma vírgula separa múltiplas opções. Por exemplo, se o dono de um ficheiro *notas.txt* unicamente para leitura quiser modificá-lo, poderá usar o comando

```
chmod u+w notas.txt
```

Ou se quiser dar a capacidade de executar um ficheiro chamado *executa.me.sff* a qualquer utilizador,

```
chmod +x executa.me.sff
```

Um exemplo mais complexo seria

```
chmod u+rw-x,g+r-wx,o-rwx exames.de.ipc1
```

ln (*Link*.) Este comando cria um **link**, ou seja, faz apontar um ficheiro para outro, como se tivéssemos dois nomes diferentes para nos referirmos ao mesmo ficheiro. É conveniente usar unicamente *links* simbólicos, por meio da opção `-s`. Um *link* simbólico representa um nome fictício para um ficheiro real, em quanto que nos *links* não simbólicos os ficheiros estão ligados fisicamente podendo criar situações confusas quando de tiver que apagar ou renomear um deles. Por exemplo, suponha que no seu sistema existe um ficheiro com uma imagem */usr/share/icons/wmaker-work.tif* que quer usar para um trabalho que está a realizar no seu directório de trabalho. Para não ter que copiar o ficheiro, ficando duas copias idênticas no mesmo sistema, pode criar um enlace simbólico:

```
ln -s /usr/share/icons/wmaker-work.tif work.tif
```

Cada vez que referir o ficheiro *work.tif* no seu directório, estará realmente a usar o ficheiro original. Se o administrador do sistema instalar uma nova versão melhorada do ficheiro *wmaker-work.tif*, o seu enlace apontará automaticamente para a versão mais recente; e poderá apagar o seu enlace sem produzir alterações no ficheiro original. Os links simbólicos aparecem indicados na primeira coluna da listagem produzida por `ls -l`, por meio de uma letra ‘l’.

cp (*Copy*.) Faz uma copia íntegra de um ficheiro:

```
cp original copia1
```

copia integralmente o ficheiro *original* para dentro de um novo designado por *copia1*. Também é possível copiar vários ficheiros

```
cp exemplo*.ps subdir
```

o caracter `*` é substituído por qualquer sequência de caracteres que produza um ficheiro que exista; neste caso em que se copiam vários ficheiros, o último argumento, *subdir*, deverá ser o nome de um directório onde serão copiados os ficheiros; pode usar também um ponto para indicar o directório corrente, quando estiver a copiar ficheiros provenientes de outro directório.

mv (Move.) Move um ficheiro ou directório para outro lugar:

```
mv /home/quim/aquitoufixe /home/toze/
```

move o ficheiro *aquitoufixe* para dentro do directório */home/toze/*

3.3 Processos

Quando um aplicativo é executado, o sistema atribui-lhe espaço em memoria, e da-lhe um identificador. O mesmo aplicativo pode correr outros programas, ou invocar outros aplicativos. A cada uma destas tarefas chamamos de **processos**. Este processo pode ocupar grande parte do processador, ou pouca. Pode até mesmo ter prioridade sobre outros processos, ou ser algo não muito importante que corre quando o processador tem tempo. Sendo assim, um processo possui inúmeras propriedades, dentro das quais podemos destacar o PID (*Process Identifier*), o utilizador que fez a invocação do processo, o comando executado e a prioridade. Um simples comando denominado por `top` deixa-nos ver, de uma forma intuitiva, os processos que estão a correr; um `top` num sistema pode resultar em algo do tipo:

```
  2:34am  up 13:25,  3 users,  load average: 0.21, 0.05, 0.02
44 processes: 42 sleeping, 2 running, 0 zombie, 0 stopped
CPU states:  5.3% user,  0.9% system,  0.0% nice, 94.0% idle
Mem:   63268K av, 59960K used,   3308K free, 19696K shrd, 2636K buff
Swap: 124988K av, 23620K used, 101368K free             29792K cached

 PID USER PRI NI  SIZE  RSS SHARE STAT LIB %CPU %MEM   TIME COMMAND
 181 root  17  0 30740  24M  6428 S      0  2.7 39.9 101:50 X
 210 root   8  0  2360 1348   600 R      0  2.7  2.1   0:02 Eterm
1866 root   5  0   768  768   596 R      0  0.5  1.2   0:01 top
 187 root   1  0  2768 1708   944 S      0  0.1  2.6   0:59 enlightenment
   1 root   0  0    60   56    44 S      0  0.0  0.0   0:04 init
   2 root   0  0     0    0     0 SW     0  0.0  0.0   0:00 kflushd
   3 root   0  0     0    0     0 SW     0  0.0  0.0   0:00 kupdate
   4 root   0  0     0    0     0 SW     0  0.0  0.0   0:00 kpiod
   5 root   0  0     0    0     0 SW     0  0.0  0.0   0:00 kswapd
  94 root   0  0   268  216   176 S      0  0.0  0.3   0:00 syslogd
  98 root   0  0   420  164   140 S      0  0.0  0.2   0:00 klogd
```

Outro comando que faz basicamente o mesmo é o `ps`:

```
  PID TTY STAT TIME COMMAND
  151   1 SW   0:00 (bash)
  152   2 SW   0:00 (mingetty)
  153   3 SW   0:00 (mingetty)
  154   4 SW   0:00 (mingetty)
  171   1 SW   0:00 (startx)
  172   1 S    0:00 tee /root/.X.err
  180   1 SW   0:00 (xinit)
  183   1 S    0:00 gnome-session
  187   1 S    0:59 (enlightenment)
  210   1 S    0:03 Eterm
 1674  p1 S    0:00 -bash
 1682  p1 S    0:04 pico COMANDOS.unix
 1867  p0 R    0:00 ps
```

Imaginemos agora que, repentinamente, um processo deixou de responder, por exemplo enquanto executava o `netscape`. De forma a poder continuar a usar o sistema operativo sem o ter de desligar, eu poderia tentar terminar o processo com o comando `kill`. Para isso, primeiro verificaria qual é o PID com o comando `ps` (ou `ps ax` que e uma forma mais alargada, mostrando-nos todos os processos que a maquina esta a correr):

```
1969   1 S    0:50 netscape
```

(Para os mais curiosos, existe uma forma de mostrar apenas as linhas que nos interessam no `ps`, que consiste em tirar partido das propriedades de *piping* e do comando `grep`. Algo do género `ps ax | grep netscape`). Depois enviaria um sinal ao programa para ele terminar:

```
kill netscape
```

E estaria tudo resolvido. Infelizmente o programa pode mesmo estar a ignorar todo o tipo de sinais. Sendo assim, a única coisa a fazer seria enviar um sinal directamente para o sistema (o *kernel* para ser mais preciso) de modo a terminar de imediato o processo. Algo do género:

```
kill -9 netscape
```

Resultaria perfeitamente[1].

Por outro lado, poderíamos querer terminar todos os processos que se chamem `netscape` faríamos o seguinte:

```
killall -9 netscape
```

3.4 Mais informações

As opções disponíveis para um determinado comando, e uma sinopse do seu uso, podem ser obtidos através de:

```
man nome-do-comando
```

o qual mostra a **página de manual** do comando ou programa.

Temos falado apenas de uns poucos comandos dos muitos existentes na *shell*. Também não temos aprofundado em outros conceitos importantes como o re-direccionamento da saída, processamento condicionado e *scripts*. Para estudar mais acerca da *shell*, consulte por exemplo a página de manual do `bash`. É conveniente também usar uma das interfaces de hipertexto para as páginas de manual, por exemplo no URL:

[1] 9 é o numero que se dá ao sinal SIGKILL.

```
http://quark.fe.up.pt/cgi-bin/man2html
```

Assim conseguirá obter uma lista por secções e por ordem alfabética, de todos os programas e comandos externos à *shell*, existentes num sistema Linux típico.

Capítulo 4

Sistema Distribuído

Os computadores Linux do DEEC formam um sistema distribuído(SD). E o que é um SD? É um conjunto de computadores interligados através de uma rede de comunicações, desempenhando papéis de clientes (utilizadores de serviços) ou servidores (fornecedores de serviços).

Actualmente o DEEC possui sete servidores distribuídos por três domínios: edifício do autocarro (EA), edifício cor-de-rosa (CR) e edifício parque-auto (PA), tal como é apresentado na figura 4.1. Estes servidores fornecem:

- espaço em disco aos utilizadores Linux do DEEC (alunos e professores), bem como o espaço em disco de alguns utilizadores irredutíveis do Windows.

- espaço em disco para software de grande volume de qualquer um dos sistemas operativos suportados pelo DEEC (Windows NT, 9x e UNIX),

- espaço em disco para documentação sobre os mais diversos assuntos de interesse da comunidade informática do DEEC,

- correio electrónico para os utilizadores do Linux do DEEC e lista de mail do Reino Linux (cortes@reinolinux.fe.up.pt),

- páginas HTML do Reino Linux (grupo de utilizadores Linux da FEUP http://reinolinux.fe.up.pt), do sistema distribuído do DEEC (http://deec00ws29.fe.up.pt), e do laboratório de redes (http://netlab.fe.up.pt).

O objectivo deste capítulo é proporcionar ao aluno um conhecimento mínimo indispensável dos sistema que irá usar durante a sua licenciatura. Esse conhecimento é de facto vital para o desenrolar óptimo das actividades informáticas em que o alunos se irá envolver.

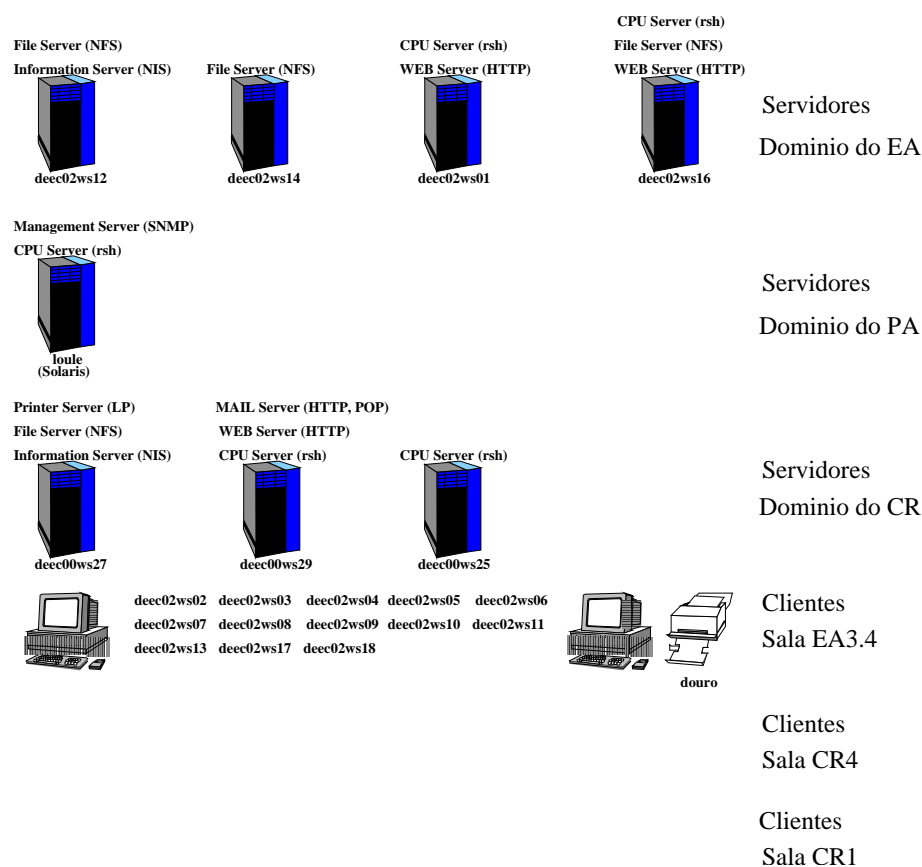



Figura 4.1: Domínios do SD do DEEC

4.1 Contas dos Utilizadores

No sistema distribuído do DEEC cada utilizador é identificado por um `login`, tem uma `password`, para (tentar) garantir que seja o único a usá-lo, e um espaço e disco para poder guardar ficheiros. A este trio em geral chama-se **conta**, i.e., o utilizador pode entrar em qualquer máquina do SD com o seu login e password acedendo sempre ao seu espaço em disco, ou área de trabalho. Para quem ainda tem dúvidas, o login, a password e a sua área de trabalho são únicos em todas as máquinas do SD

Esta uniformidade é conseguida devido à existência do serviço de informação NIS[1]. O computador servidor principal deste serviço é conhecido por `Servidor Mestre` é onde toda a informação do sistema distribuído é introduzida, nomeadamente informação sobre a conta do utilizador,.

Como o SD do DEEC se espalha por vários edifícios e como tem um número elevado de estações clientes, foram instalados diversos `Servidores Escravos` nos vários domínios do SD. Estes servidores escravos contêm uma réplica exacta

[1]NIS – Network Information System

da informação presente no servidor mestre. Quando um utilizador faz um **login** numa máquina, esta pergunta ao servidor de NIS mais próximo se a password está correcta e, em caso afirmativo, pede também a localização física (servidor remoto, sistema de ficheiros) da área de trabalho. De seguida monta a área de trabalho do utilizador no directório */home* da máquina em questão para o utilizador a poder usar. Conclusão, se um utilizador escrever um texto que guarda na seu espaço em disco de uma estação de trabalho, quando quiser continuar o texto não necessita de entrar exactamente na mesma estação de trabalho, pois o seu espaço em disco é único e aparecerá em qualquer computador em que entre.

Como dissemos atrás as áreas de trabalho dos utilizadores aparecem montadas em */home*. Sempre que alguma operação envolva ficheiros da sua área de trabalho, esta é montada automaticamente. Para se verificar tal facto podem proceder a uma experiência utilizando o comando `ls`:

```
ls -l /home
```

este comando vai vos indicar o que está em */home*. Podem ver as contas que estão montadas neste computador e uma delas é a vossa. No meu caso a resposta do `ls` foi:

```
pbraga@
```

de seguida listar a conta de um utilizador qualquer:

```
ls  ~caloiroedu
```

o **til** imediatamente anterior ao nome do utilizador serve para indicar o espaço em disco de um dado utilizador. Em **UNIX** todos os utilizadores têm um espaço em disco que pode estar em qualquer ponto da hierarquia de ficheiros. Assim, utilizando o **til**, poupamos trabalho de pesquisa e temos a certeza de que estamos a indicar a posição certa – a conta do utilizador. Neste caso usei o utilizador `caloiroedu`. O importante para esta experiência é que a conta deste utilizador não esteja ainda em */home*.

Por último, execute outra vez o comando `ls` de */home*:

```
ls /home
```

o meu resultado foi o seguinte:

```
caloiroedu@  pbraga@
```

poderá ver que apareceu a conta do utilizador `caloiroedu` em */home*.

Para saber a localização física da minha conta posso executar o comando `ls -l /home`. Como exemplo de resposta:

```
total 4
lrwxrwxrwx 1 root  29 Sep 10 11:59 ee99777 -> /amd/deec00
ws27/e99-1/ee99777/
lrwxrwxrwx 1 root  32 Sep 10 11:29 pbraga -> /amd/deec02w
s12/usr/pedro/pbraga/
```

olhando para estas duas linhas podem ver que o utilizador `ee99777` tem a sua conta na máquina `deec00ws27` na *file-system* `e99-1`. i.e., o nome da máquina vem sempre a seguir a */amd* e depois o que fica entre o nome da máquina e do utilizador é a *file-system*.

4.2 Sistema de Ficheiros Distribuídos

O serviço principal do SD do DEEC é de facto o sistema de ficheiros distribuídos, que proporciona o aparecimento da área de trabalho de cada utilizador em qualquer máquina de uma forma quase transparente. Para além disso, este serviço é ainda importante para outros sistema de ficheiros, como sejam aqueles contendo espaços correio electrónico, software e documentação.

Um sistema de ficheiros distribuídos pode ser dividido em duas partes principais: a montagem de sistemas de ficheiros remotos e o acesso aos ficheiros. No caso do SD do DEEC usaram-se dois protocolos para esse efeito: Mount e NFS.

4.2.1 Mount

Por esta altura devem estar a perguntar a vós próprios:

> Mas afinal o que vem a ser isto de montar? Tem a ver com cavalos ou bicicletas?

Quando se coloca uma disquete num **PC-DOS**, passamos a ter acesso ao seu sistema de ficheiros usando para isso a designação «**A:**». Depois, se trocarmos de disquete, passamos a ter um sistema de ficheiros diferente na mesma designação lógica («**A:**»). Ora, o que se fez foi montar a primeira disquete em «**A:**» para de seguida a desmontarmos e montarmos a segunda. Portanto:

> Montar é associar uma *file-system* física a um ponto da estrutura lógica de ficheiros do Sistema Operativo.

4.2.2 NFS

Como já referimos atrás, montar uma *file-system* é associa-la a um ponto lógico da estrutura de ficheiros. Mas como é que se consegue associar uma *file-system* que

se encontra fisicamente noutro computador à estrutura de ficheiros do computador local?

Existe um serviço cujo nome é NFS[2] que, através da rede de comunicações, fornece a um computador acesso a uma *file-system* remota de modo transparente para os restantes processos. Ao computador que disponibiliza[3] a *file-system* chama-se de `servidor de NFS` e de `cliente de NFS` ao computador que utiliza esse serviço. A figura 4.2 exemplifica a arquitectura deste serviço, mostrando servidores exportando sistemas de ficheiros, e clientes montando-os constituindo árvores de directórios baseadas em sistemas de ficheiros locais e remotos.

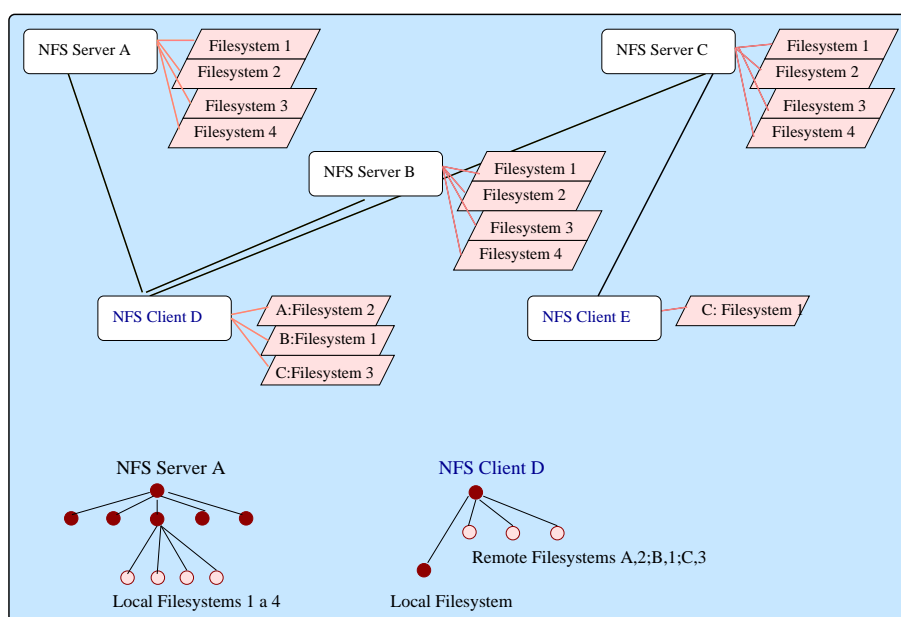



Figura 4.2: Sistema de ficheiros distribuídos baseado em NFS.

Assim numa estação de trabalho existem diversos sistemas de ficheiros, uns locais outros remotos, para o verificar execute o comando `df` num terminal X:

```
Filesystem        1024-blocks    Used   Avail. Capac. Mounted on
/dev/hda1               97556   63615   28903    69%  /
/dev/hda3             1190046  513134  615424    45%  /usr
deec00ws29:/var/spool/mail
                      3842436 1269507 2374125    35%  /var/spool/mail
/dev/hda4             1127638  686665  382711    64%  /home14
deec02ws12:/homestaff
                      1018298  777271  188416    80%  /amd/deec02ws12/homestaff
deec02ws14:/home14    1127638  686665  382711    64%  /amd/deec02ws14/home14
deec00ws27:/homestaff2
                      3648613 1118884 2340983    32%  /amd/deec00ws27/homestaff2
```

[2]NFS – Network File System

[3]O termo técnico usado neste caso é *exportar* a *file-system*

Do resultado do comando apercebemo-mos que as três partições do disco local:

- */dev/hda1*,
- */dev/hda2*,
- */dev/hda4*,

foram montadas nos directórios

- */*
- */usr*
- */home14*

Por outro lado os quatro sistemas de ficheiros remotos:

- */var/spool/mail*,
- */homestaff*,
- */home14*,
- */homestaff2*),

respectivamente dos servidores deec00ws29, deec02ws12, deec02ws14 e deec00ws27, foram montados nos directórios:

- */var/spool/mail*,
- */amd/deec02ws12/homestaff*,
- */amd/deec02ws14/home14*,
- */amd/deec00ws27/homestaff2*.

Para além disso ainda nos é possível ver o tamanho total dos sistema de ficheiros em blocos de 1kB, bem como o espaço usado e disponível.

4.3 Serviço de Impressão Distribuída

O serviço de impressão distribuída concebido no DEEC, e que se encontra em fase de implementação, é composto por diversos elementos:

- impressoras,
- servidores de impressão,
- base de dados,
- filtros
- filas de espera.

Quando o utilizador pede para imprimir um ficheiro tem que especificar o nome da impressora e o nome do ficheiro. Os trabalhos de impressão são todos enviados para o servidor de impressão que analisa o pedido e os faz passar por filtros para, por exemplo, contar o número de paginas a imprimir, verificar na Base de Dados se a impressão em curso não excede a cota do utilizador, ou garantir que o ficheiros está no formato correcto (PostScript). Na situação do trabalho de impressão ser aceite ele é enviado para uma fila de espera correspondente ao nome lógico da impressora usada . De facto, uma impressora física pode ter mais que uma impressora lógica, por exemplo, para imprimir em face simples ou dupla [4].

4.3.1 Imprimir em Linux

No SD do DEEC para se imprimir um ficheiro a partir do sistema operativo Linux deve-se utilizar o comando:

```
lpr -P<impressora> <ficheiro>
```

em que <impressora> é o nome lógico associado a uma impressora física onde se quer imprimir e <ficheiro> é o nome do ficheiro a imprimir. Em princípio esse ficheiro terá que estar no formato PostScript(PS), e os nomes dos ficheiros têm, normalmente, «`.ps`» no fim.

As impressoras localizadas em salas de alunos estão sujeitas a uma gestão de cotas baseada no número de impressões por utilizador. Para verificar o estado das impressoras e da sua cota o utilizador pode consultar uma página Web do SD do DEEC dedicada a esse fim:

```
http://deec00ws29.fe.up.pt/impressora/
```

[4] Como é o caso da impressora da sala EA3.4.

4.4 Correio Electrónico

No SD do DEEC todas as máquinas Linux funcionam como servidores de mail, daí que é possível enviar e receber mensagens a partir de qualquer uma destas máquinas. Contudo, como a maioria das máquinas tem `dual boot` não é garantido que a máquina esteja permanentemente em Linux e como tal que o servidor de mail esteja em funcionamento. Assim, as máquinas que de momento estão permanentemente ligadas em Linux e às quais todos os utilizadores têm acesso são: deec00ws25, deec00ws29, deec02ws16 e deec02ws01.

Nas máquinas Linux do SD do DEEC quando chega uma mensagem para um utilizador, esta é guardada em */var/spool/mail/*. Para garantir que o utilizador não tem que andar a ler as mensagens a si dirigidas em todas as máquinas do SD do DEEC, o directório */var/spool/mail* de todas as máquinas é importada da máquina deec00ws29, servidor de mail oficial do SD do DEEC. Se repetir o comando `df` irá encontrar uma linha no resultado que lhe diz que o `spool` do correio electrónico provém de facto da máquina deeec00ws29.

```
Filesystem                1024-blocks  Used   Avail. Capac. Mounted on
deec00ws29:/var/spool/mail 3842436  1271694 2371938  35%   /var/spool/mail
```

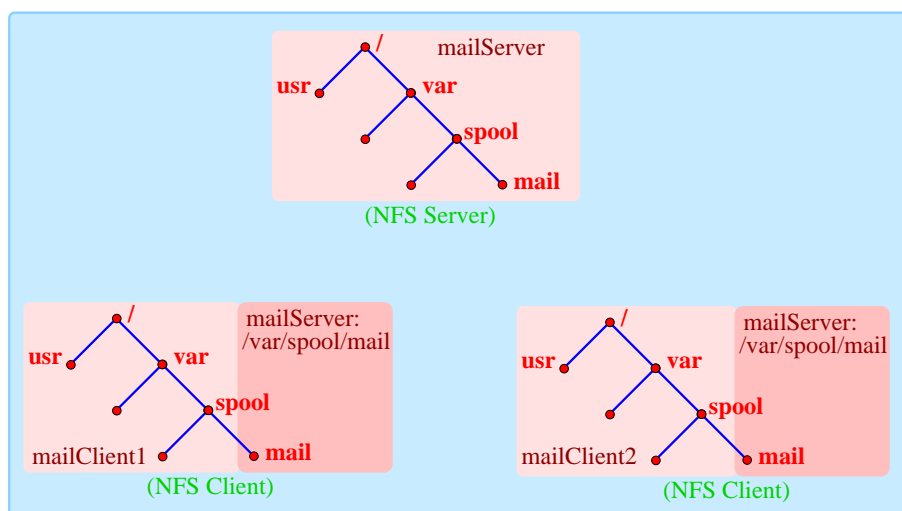



Figura 4.3: Utilização conjunta do sub-sistema de correio electrónico e sistema de ficheiros distribuídos.

A figura 4.3 dá uma ideia dos princípios usados para colocar o mesmo espaço em disco dedicado ao `spool mail` em todas as máquinas. Note-se que o sistema de mail do SD do DEEC é suportado fortemente nos mecanismos de sistemas de ficheiros distribuídos, neste caso particular no NFS.

4.4.1 Leitores de Correio

Em Linux a forma mais simples de ler o correio electrónico recorre ao comando `elm`, que lhe proporciona uma interface de texto ligeira e de fácil uso para ler mensagens simples. O programa `elm` sempre que arranca vai ler no */var/spool/mail* o ficheiro correspondente ao utilizador e permite ler o correio directamente sobre o sistema de ficheiros usado pelo serviço de mail */var/spool/mail*.

O comando UNIX `from` permite-lhe ver as mensagens que estão na sua caixa de correio sem ter que invocar um `mailer`[5]

4.5 Serviços informáticos do CICA

O CICA (Centro de Informática Prof. Correia de Araújo) é o centro informático da Faculdade de Engenharia e tem como funções:

- Gerir a rede e as máquinas que a compõem e ser responsável pela manutenção e reparação da mesma.
- Tem à disposição dos alunos duas salas de computadores que se localizam no edifício central da faculdade.
- Administra a rede WinNT.
- É responsável por introduzir na rede os programas/aplicações exigidas para o bom funcionamento das aulas.
- Possui um serviço de reprografia de grande dimensão.
- É responsável pelo apoio técnico aos utilizadores da FEUPNet, possuindo também um gabinete de ajuda e esclarecimento de dúvidas (*helpdesk*).
- Tem disponível material para produção de slides e gravação de CD-ROMs, mediante marcação.
- Executa impressões de vários tipos/qualidades.
- Gere e administra o SiFEUP (ver a próxima secção).
- Garante o acesso remoto à FEUPNet.
- Fornece listas de distribuição via e-mail.

Toda informação acerca dos serviços prestados pelo CICA poderá ser encontrada na sua página web em `http://www.fe.up.pt/cica3w`.

[5] Leitor de correio electrónico no calão informático.

Devido ao crescimento do parque informático da Faculdade, muitos serviços têm vindo a ser gradualmente descentralizados do CICA, nomeadamente, a disponibilização de salas de PC's no DEEC, como foi descrito nas secções anteriores. No entanto, todos os alunos da FEUP têm direito a ter uma conta no CICA (no caso das licenciaturas em Electrotecnia e Informática o *username* será o mesmo usado nos sistemas do DEEC, mas com um password diferente), que lhes permitirá ter acesso aos serviços do CICA.

4.5.1 Acesso ao correio electrónico do CICA

Com a conta do CICA o aluno fica com um endereço de mail da forma:

`utilizador@fe.up.pt`

Para se poder ver ou enviar correio a partir da conta do CICA pode-se usar o `Netscape` depois de correctamente configurado. Para configurar o `Netscape` deve-se escolher as seguintes opções «`edit`» e «`preferences`» (ver fig. 4.4).

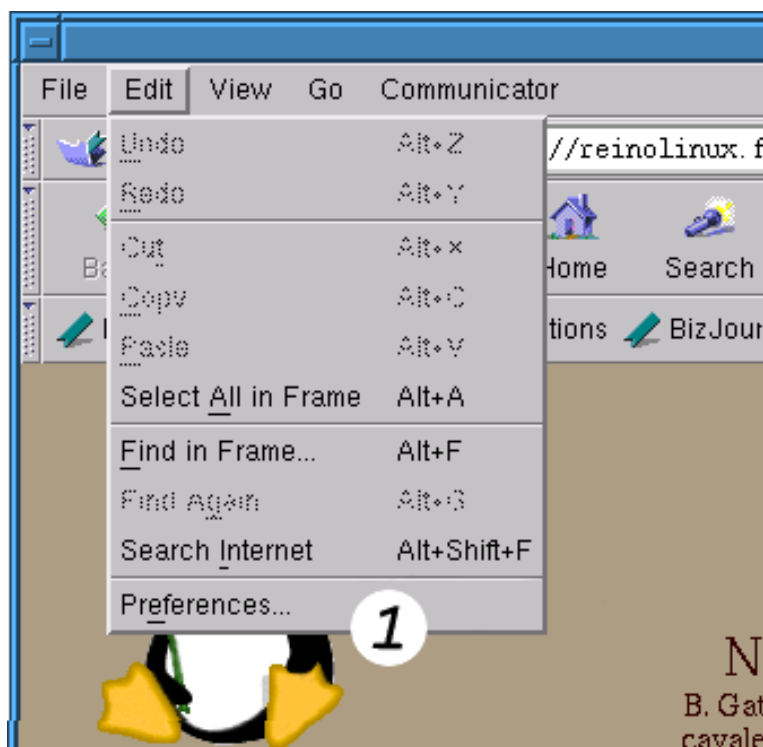



Figura 4.4: Menu `preferences` do Netscape

Depois deve-se escolher a opção «`Mail & Newsgroups`» que fornece várias escolhas. A primeira coisa a inserir, dentro de `Mail & Newsgroups`, é a identificação (fig. 4.5).

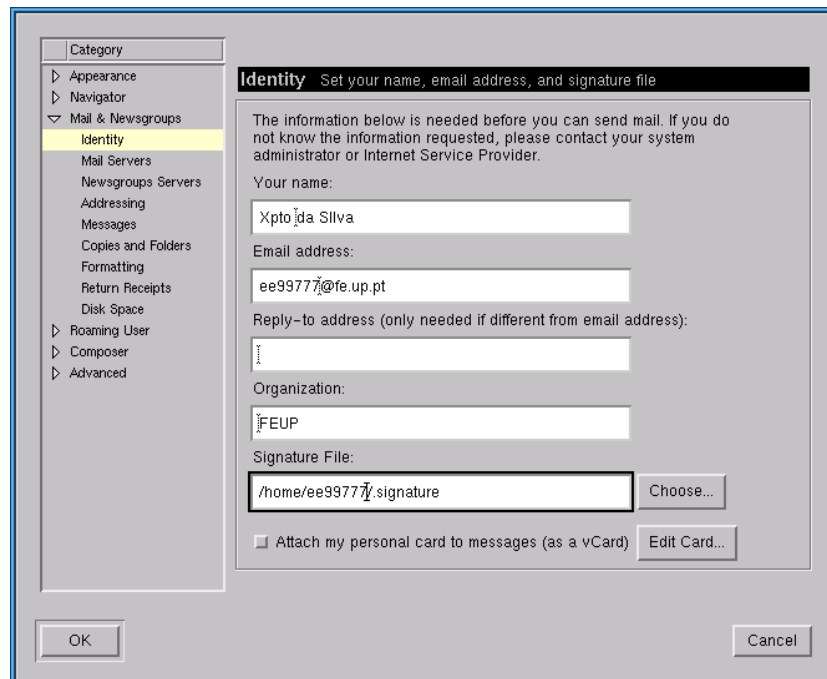



Figura 4.5: Formulário para inserir identificação

O `Email address` é o mail que o destinatário verá, e para onde irá responder caso não se preencha o campo `Reply-to address`. A `Signature file` é um ficheiro de texto que o Netscape insere automaticamente no fim de um mail sempre que vamos começar a escrever um.

Depois de inserir a identificação tem que se informar o Netscape sobre o servidor onde temos o mail. A opção «`Mail Servers`» deve ser seleccionada (fig 4.6).

Na segunda secção onde se define `Outgoing Mail Server`, deve-se escrever como servidor: "`alf.fe.up.pt`"e como `user name` o login do CICA[6].

A `Local mail directory` é o directório, na nossa conta local, onde o Netscape vai colocar as mensagens que queremos guardar e as que acaba de receber.

Voltando ao início desta janela, na primeira secção `Incoming Mail Servers`, deve-se adicionar[7] o servidor (fig. 4.7) com o login correcto.

4.5.2 O SiFEUP

O Sistema de Informações da FEUP (SiFEUP) é uma base de dados automatizada que permite aos alunos, funcionários ou professores aceder a numerosas informações a seu respeito (ou, no caso dos professores, também aceder a informação sobre os alunos). Através da pesquisa pelo SiFEUP, e sabendo o nome ou número de

[6]normalmente este login é igual ao do DEEC

[7]se já houver um definido deve-se editar e não adicionar.

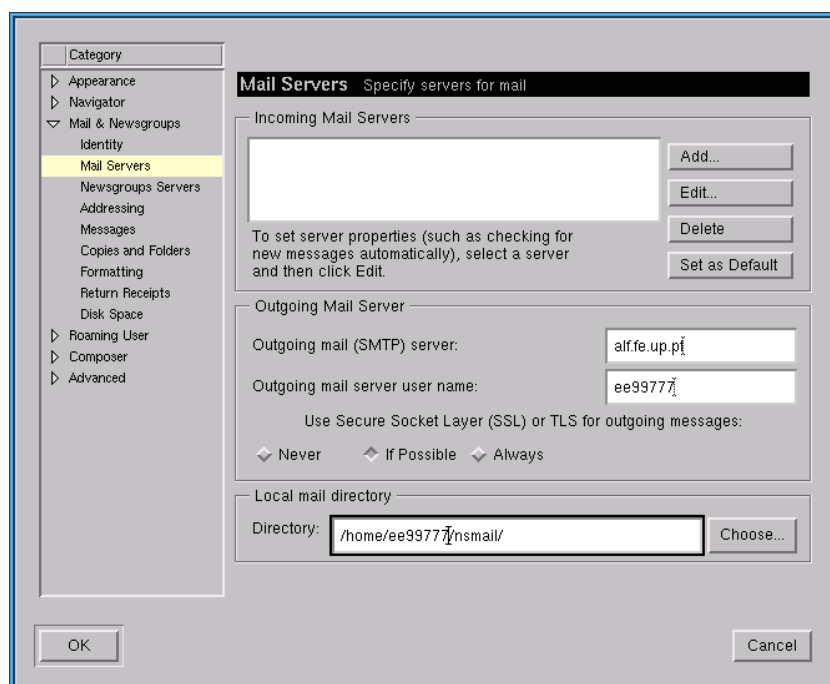



Figura 4.6: Definir Servidores de mail

aluno/funcionário, poder-se-á encontrar qualquer aluno, funcionário ou professor da FEUP. Nessa página pode-se verificar o ano em que o aluno está inscrito, o seu horário, o seu e-mail, a sua homepage, o seu número de telefone de contacto, fotografia, etc. [8]

O SiFEUP permite ainda o acesso à sua ficha de aluno, com informação de todas as cadeiras em que está/esteve inscrito, e as respectivas classificações obtidas, bem como a média actual do curso. Poderá ainda, em alguns cursos efectuar a sua inscrição nas turmas através do SiFEUP.

4.6 Acesso remoto à FEUPNet

Existem duas formas de aceder remotamente à FEUPNet, transformando assim o seu computador de casa numa máquina temporariamente pertencente à rede da Faculdade de Engenharia.

Uma opção será assinar o protocolo RCU (Redes de Comunicação para Universitários) da Portugal Telecom que concede preços mais favoráveis nas ligações de dados à rede da FEUP via RDIS. A adesão a este protocolo possui, no entanto, uma lista de espera bastante longa, o que o torna bastante menos apetecível.

A segunda opção de ligação à FEUPNet será a chamada opção **normal**, que

[8]Para um aluno poder retirar/alterar estes elementos da sua página de aluno, terá que obter uma outra password para SiFEUP, junto do pessoal do CICA.

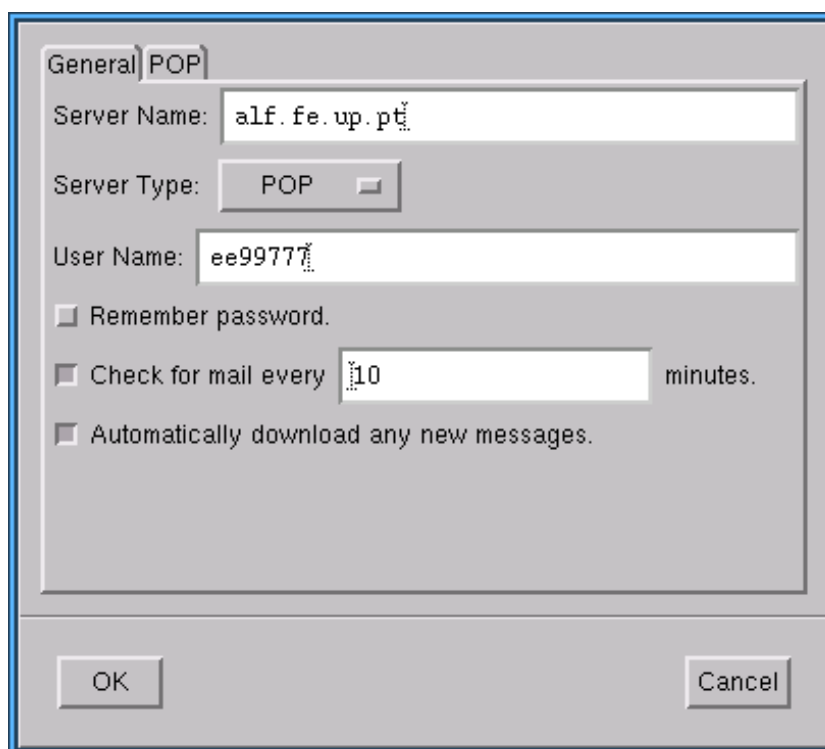



Figura 4.7: Definir propriedades do servidor de mail

consiste em ligar via modem analógico ou placa RDIS para um dos números de atendimento de dados do CICA, essa chamada será dirigida para a máquina que atende as chamadas dos alunos provenientes das suas casas (*tcpgate.fe.up.pt*), e, tendo o computador propriamente configurado em casa, poderá começar a trabalhar, quer na rede interna da FEUP, quer na internet, como se estivesse num qualquer outro terminal da Faculdade ou do DEEC. Para isso basta pedir no CICA (ou através da página web http://www.fe.up.pt/cica3w/formularios/tcpgate.html) um login na maquina que atende as chamadas telefónicas (tcpgate).

Capítulo 5

Processamento de texto

Existem duas metodologias diferentes para a produção de documentos impressos usando o computador. Os processadores de texto baseados na primeira metodologia, designada de *Wysiwyg* (*What you see is what you get*), pretendem apresentar no ecrã o documento tal como será impresso na página. A segunda metodologia consiste na criação de um documento fonte com determinados comandos que após serem processados produzem o documento na forma na qual será impresso.

Alguns sistemas *Wysiwyg* disponíveis em Linux são o StarOffice, Lyx e WordPerfect; na segunda classe de programas os mais usados são Tex, LaTeX e Troff. À primeira vista pode parecer mais fácil e rápido usar sistemas *Wysiwyg* por não precisarem do processamento de nenhum ficheiro fonte, mas na realidade existe um ficheiro fonte que está constantemente a ser processado e assim, estes sistemas costumam ser muito mais lentos, ocupam mais espaço, e dependem de um sistema operativo em particular. Num sistema como o LaTeX, a criação do documento fonte é rápida (uma vez se tenha algum conhecimento mínimo da sua sintaxe) mas quando o utilizador quer visualizar o ficheiro como será impresso, o processamento do documento pode demorar alguns segundos; é de salientar que no ambiente gráfico é possível ter duas janelas, uma em modo texto onde se está a processar o ficheiro fonte, e outra em modo gráfico que apressenta o ficheiro LaTeX processado, na forma como será impresso.

Neste capítulo introduziremos o sistema LaTeX, por acharmos que possui muitas vantagens sobre os sistemas *Wysiwyg*. A primeira vantagem é que em LaTeX o autor identifica mais facilmente a estrutura lógica do documento (divisão em capítulos, secções, etc) sem ser incomodado com pormenores de estética e apresentação final do documento (tipos de letra, tamanhos, etc). Nos sistemas *Wysiwyg*, é difícil manter a consistência do documento especialmente quando este tem várias páginas. Por exemplo, suponha que no texto decide fazer ênfase em certas palavras que aparecem em línguas estrangeiras ou que definem novos conceitos, usando diferentes tipos de letra para um caso ou outro; num sistema *Wysiwyg*, será a sua responsabilidade lembrar o tipo de letra seleccionado para cada caso; em LaTeX pode definir um par de comandos com quaisquer nomes como `\est` e `\definicao`

que usará cada vez que apareça uma palavra em lingua estrangeira ou uma definição: `\est{palavra estangeira}` `\definicao{nome a definir}`; esta metodologia tem a vantagem de lembrar ao autor do porquê o texto aparece num tipo diferente de letra; também permite mudar facilmente o tipo de letra usado por exemplo em todas as palavras estrangeiras que apareçam no texto. Num sistema *Wysiwyg* isso implicaria uma revisão cuidadosa para modificar cada palavra estrangeira; em LaTeX bastaria uma simples re-definição do comando `\est` no início do documento.

Outra vantagem dos sistemas que trabalham por comandos é que registam a sequência usada para obter determinado resultado, sendo fácil de ser copiada e modificada por outros autores ou pelo próprio autor. Nos sistemas *Wysiwyg* realmente estão a ser usados comandos cada vez que se carrega num botão ou menu; mas a sequência de opções do menu ou botões seleccionados que deram origem a um determinado documento podem ser esquecidos ou difíceis de reproduzir. Um documento produzido com um sistema *Wysiwyg* é como uma pintura que pode ser difícil de copiar por alguém que não tenha a mesma perícia do seu autor, e inclusivamente pelo própio autor quando não estiver no mesmo estado de inspiração; seguindo a mesma analogia, a versão LaTeX da «pintura» seria uma série de comandos que ilustram perfeitamente os passos seguidos para a sua produção; copiando um estilo criado em LaTeX por um artista gráfico, qualquer pessoa pode produzir um livro com a mesma perfeição gráfica, sem precisar de ter qualquer talento artístico[1]; o criador do documento pode ser até um programa de computador, que utilizando o mesmo estilo e alterando alguma informação proveniente de uma base de dados, produz vários documentos com o mesmo formato gráfico e diferentes conteúdos.

5.1 Documento tipo

A forma mais fácil de aprender LaTeX é olhando para um exemplo simples que ilustre as principais partes que deverá ter um documento. Consideremos um ficheiro *exemplo1.tex* que contém o seguinte documento LaTeX:

exemplo1.tex

```
\documentclass{article}
\usepackage[latin1]{inputenc}
\begin{document}

\section{Introducão}
Este é um exemplo simples de um documento
criado usando LaTeX. % este é um comentário
\section{Linhas e parágrafos}
Eis    a    segunda secção do documento.
```

[1]Um exemplo de documento criado com LaTeX é este próprio manual.

```
O texto será sempre justificado à direita,
hifenizando palavras se chegar a ser necessário.
Para separar parágrafos será necessário deixar
pelo menos uma linha em branco.
\end{document}
```

Qualquer documento deve aparecer entre os comandos `\begin{document}` e `\end{document}`. Antes do início do documento é necessário definir o estilo do documento, por meio do comando `documentclass`. Neste exemplo o estilo usado para o documento é `article`. Outros estilos pré-definidos, usados com frequência, são: book, report e letter.

O documento do exemplo tem duas secções («Introdução» e «Linhas e parágrafos»). Os nomes das secções serão escritos em letra de tamanho maior e em caracteres negros, seguindo o estilo definido para o documento. A numeração das secções corresponde ao interpretador do LaTeX, e não ao autor, seguindo as normas no estilo pré-definido.

Alguns caracteres em LaTeX não são interpretados como parte do texto, mas como uma indicação para que o interpretador do LaTeX entre em algum modo especial; por exemplo o símbolo % indica que o que vem na continuação da linha é um comentário que será ignorado, como no caso do texto que aparece no fim da introdução. Os outros caracteres especiais em LaTeX são: `{ } & # $ _ ~ ^`. Quando algum caracter especial aparece no ficheiro fonte, não é copiado para o ficheiro de saída, mas é interpretado pelo programa `latex` em alguma forma especial. Para obter um destes caracteres especiais no ficheiro de saida, deverá ser precedido por uma barra inclinada para a esquerda ($\backslash$); no texto impresso desaparece a barra, ficando unicamente o caracter especial.

O próprio símbolo $\backslash$ que acabamos de referir é um pouco mais difícil de obter no texto: `$\backslash$`, mas como não é usado com muita frequência, foi adoptado em LaTeX para indicar os nomes de comandos. Qualquer palavra a começar por esse caracter é interpretada como um comando que executa alguma acção ou define alguma coisa. O operando (ou operandos) sobre o qual actua o comando, costumam vir a seguir ao operador, e dentro dos caracteres especiais: { e }.

O símbolo $ usado acima para produzir $\backslash$, é um dos caracteres especiais; tudo o que estiver entre dois símbolos $ é interpretado como uma fórmula matemática. Alguns caracteres como as letras gregas ou o próprio «backslash» fazem parte da tabela de símbolos matemáticos e não do alfabeto romano, e por isso é que é preciso inserir o comando que produz o backslash dentro de dois caracteres $.

O comando `{\usepackage[latin1]{inputenc}` no início do ficheiro é para permitir o uso de caracteres acentuados no ficheiro fonte.

5.2 Processamento de ficheiros LaTeX

Uma vez criado um documento LaTeX, como por exemplo o ficheiro *exemplo1.tex* na secção anterior, será preciso processá-lo para produzir o ficheiro como será impresso. Normalmente será usado o comando

`latex` *exemplo1.tex*

para produzir um ficheiro *exemplo1.dvi* que será logo transformado em comandos para uma determinada impressora, usando filtros como `dvips` e `dvilj`. Existem muitas outras possibilidades, para além da criação do ficheiro `dvi`, segundo o gosto do autor e o fim ao qual se destina o ficheiro final (Figura 5.1); por exemplo, o comando

```
pdflatex exemplo1.tex
```

produz um ficheiro *exemplo1.pdf*.

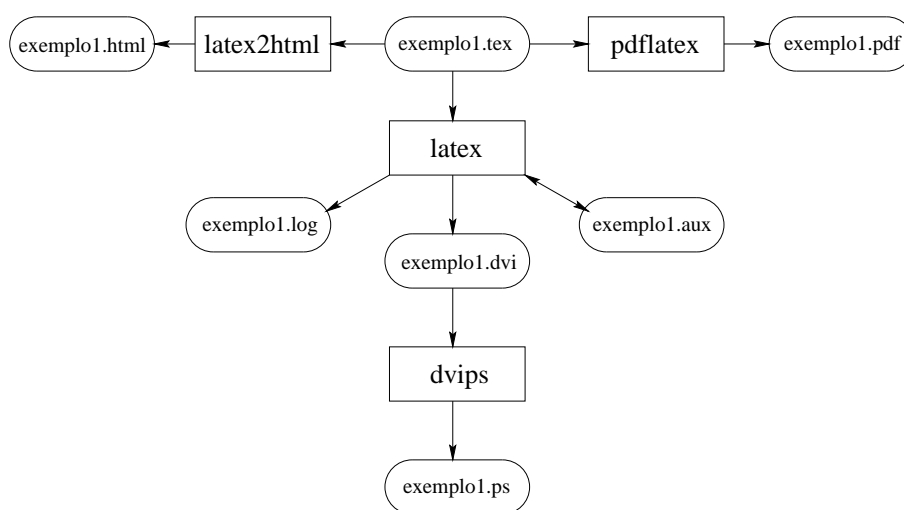



Figura 5.1: Processamento do ficheiro fonte *exemplo.tex* para produzir ficheiros em formatos PostScript, pdf e html.

Para produzir um ficheiro *exemplo1.html* em formato html, a partir do ficheiro *exemplo1.tex*, pode usar-se o seguinte comando

```
latex2html -split 0 -info 0 -no_subdir exemplo1.tex
```

a opção `split 0` é necessária para que não sejam separadas as duas secções do documento em dois ficheiros separados (já que neste caso as secções são tão pequenas); como já não serão necessários botões de navegação, pode-se usar também a

opção `no_navigation` para os suprimir. A opção `info 0` é para suprimir alguma informação sobre o programa `latex2html` que costuma ser incluída no fim do ficheiro, e a opção `no_subdir` evita a criação de um sub-directório *exemplo1* onde normalmente seria colocado o ficheiro resultante *exemplo1.html*. No caso de documentos mais complexos, com várias secções, o comando `latex2html` sem opções produz um sub-directório com vários ficheiros `html`, que permitem uma fácil navegação através da secções do documento (em ficheiros separados).

Seja qual for o tipo de ficheiro que se pretende produzir a partir do ficheiro fonte *exemplo1.tex*, convém sempre usar o comando `latex` para verificar que o ficheiro fonte é um ficheiro LaTeX válido (sem erros de sintaxe). Por exemplo, modifiquemos a linha que diz `\section{Introdução}`, no exemplo 1, para: `\seccao{Introdução}`. Quando tentarmos processar o ficheiro fonte com o comando `latex exemplo1` (a extensão `.tex` pode ser omitida), o resultado será o seguinte

```
! Undefined control sequence.
l.5 \seccao
           {Introduc^^e3o}
?
```

Este resultado indica que o programa latex identificou `\seccao`, na alínea 5 do ficheiro, como um comando desconhecido. O sinal de interrogação corresponde a que o programa latex está a espera de uma decisão por parte do utilizador: abortar, substituir o comando por outro que esteja correcto, obter uma explicação melhor da causa do erro, etc. Se respondermos com outro sinal de interrogação, obteremos a lista das possíveis acções:

```
Type <return> to proceed,
S to scroll future error~messages,
R to run without stopping, Q to run quietly,
I to insert something, E to edit your file,
1 ... or 9 to ignore the next 1 to 9 tokens of input,
H for help, X to quit.
```

se, por exemplo, respondermos com o comando `h`, obteremos a seguinte explicação:

```
The control sequence at the end of the top line
of your error message was never \def'ed. If you have
misspelled it (e.g., `\hobx'), type `I' and the
correct spelling (e.g., `I\hbox'). Otherwise just
```

```
continue, and I'll forget about whatever was
undefined.
```

Isto quer dizer que podemos escrever `i\section` e o ficheiro será processado dando origem a um ficheiro *exemplo1.dvi* que pode ser visualizado por exemplo com o `xdvi` (ver Capítulo 1). No entanto, o comando `i\section` não modificará o ficheiro fonte, *exemplo1.tex*, o qual continua errado. Para corrigir o ficheiro pode ser usado o comando `e` o qual abre um editor de texto (normalmente o editor `vi`) no local onde aconteceu o erro, ou se preferirmos usar outro editor usaremos o comando `x` para sair do programa latex. Em ambiente gráfico um método de trabalho bastante conveniente consiste em trabalhar com 3 janelas; uma delas terá o ficheiro fonte dentro dum editor de texto, a segunda terá um *shell* Linux que permita dar o comando `latex` *exemplo1*, e a terceira terá o ficheiro *exemplo1.dvi* visualizado com o `xdvi` ou outro visualizador de ficheiros `dvi`. Quando aparece um erro no ficheiro fonte, corrige-se o erro no editor de texto, usa-se a seta que aponta para cima no *shell*, para repetir o último comando (que será `latex exemplo1`), e o resultado pode ser actualizado na janela do `xdvi`, simplesmente com tornar essa janela activa.

Neste momento podemos referir uma funcionalidade do LaTeX. Suponha que descobrimos que o comando `section` foi repetidamente escrito erroneamente como `seccao`; em vez de corrigir as várias ocorrências do erro, podemos introduzir a seguinte definição no **preâmbulo** (antes do comando `\begin{document}`):

```
\newcommand{\seccao}[1]{\section{#1}}
```

Assim, definimos o comando `\seccao` como um sinónimo de `\section`; o argumento 1 entre parênteses quadrados indica o número de operandos do comando definido (aquilo que virá a seguir ao nome do comando, entre chaves, quando o comando for usado no texto), e o #1 indica o lugar onde entrará o primeiro operando.

5.3 Outros comandos do LaTeX

Para introduzir outros comandos do LaTeX, vamos apresentar um segundo exemplo mais complexo, que depois discutiremos em pormenor. O ficheiro fonte do segundo exemplo é o ficheiro *exemplo2.tex* que se segue

exemplo2.tex

```
\documentclass[a4paper,12pt]{article}
\usepackage[portuges]{babel}
\usepackage[latin1]{inputenc}
\usepackage[dvips]{graphicx}
```

```
\begin{document}
\title{Segundo Exemplo}
\author{J. Villate}
\maketitle
\begin{enumerate}
\item (4 valores) Para medir o coeficiente de
  atrito estático entre um bloco e um disco,
  fez-se rodar o disco com uma aceleração angular
  $\alpha = 5$ rad/s$^2$ constante. O disco parte
  do repouso em $t = 0$ e no instante $t = 0,82$ s
  o bloco começa a derrapar sobre o disco.
  \begin{enumerate}
  \item Desenhe as forças que actuam sobre o bloco
    (visto de cima e de lado).

  \item Indique a direcção e sentido da aceleração
    do bloco antes de derrapar (visto de cima e de
    lado).

  \item Calcule o coeficiente de atrito estático
    entre o bloco e o disco
  \end{enumerate}
  \begin{center}
    \includegraphics{disco.ps}
  \end{center}
  \textbf{Resolução}. (\emph{a}) Enquanto o bloco
  não começar a derrapar ($r = 8$ cm, constante),
  a sua aceleração tangencial será
  \begin{equation}
    a_t = \alpha r = 5\times 8 \;\frac{cm}{s^2}
    = 40 \;\frac{cm}{s^2}
  \end{equation}
\end{enumerate}
\end{document}
```

O ficheiro obtido depois de processar *exemplo2.tex* com os programas `latex` e `dvips` aparece na figura 5.2. Na definição do tipo de documento, na primeira alínea de *exemplo2.tex*, a lista de palavras entre parênteses quadrados são **opções** para o pacote `article`. A opção `a4paper` é para mudar o formato da página, que por omissão seria tamanho carta; a opção `12pt` é para mudar o tamanho da fonte do texto para 12 pontos (por omissão seria 10 pontos); todas as dimensões,

Segundo Exemplo

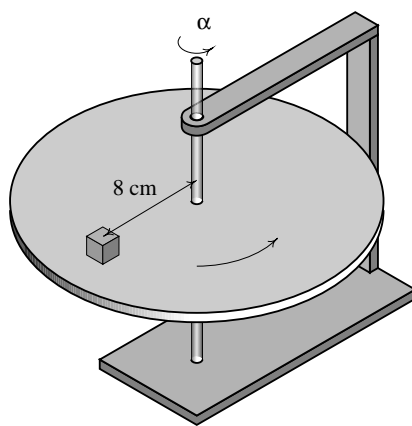
J. Villate

10 de Setembro de 1999

1. (4 valores) Para medir o coeficiente de atrito estático entre um bloco e um disco, fez-se rodar o disco com uma aceleração angular $\alpha = 5$ rad/s^2 constante. O disco parte do repouso em $t = 0$ e no instante $t = 0,82$ s o bloco começa a derrapar sobre o disco.

   (a) Desenhe as forças que actuam sobre o bloco (visto de cima e de lado).

   (b) Indique a direcção e sentido da aceleração do bloco antes de derrapar (visto de cima e de lado).

   (c) Calcule o coeficiente de atrito estático entre o bloco e o disco

Resolução. (a) Enquanto o bloco não começar a derrapar ($r = 8$ cm, constante), a sua aceleração tangencial será

$$a_t = \alpha r = 5 \times 8 \frac{cm}{s^2} = 40 \frac{cm}{s^2} \qquad (1)$$

Figura 5.2: Versão final do exemplo 2, obtida a partir do ficheiro fonte *exemplo2.tex*

como por exemplo o tamanho dos títulos e o espaço entre linhas, serão aumentadas proporcionalmente.

Foram introduzidos vários comandos novos no exemplo 2; os comandos `usepackage` no preâmbulo do documento, são usados para carregar pacotes que estendem ou modificam a versão original do LaTeX. O pacote `inputenc`, como já vimos no exemplo 1, permite o uso de caracteres acentuados no ficheiro fonte; a opção `latin1` dada ao pacote `inputenc` indica que o código usado para representar caracteres acentuados é o código *iso-latin1*. O pacote `babel` introduz suporte para várias línguas dentro do LaTeX; a opção *portuges* (repare que «portuges» não leva «u» já que assim foi definida pelo autor do pacote) faz com que sejam usadas as regras de hifenização em português, e traduz algumas palavras que são introduzidas automáticamente pelo programa `latex` como, por exemplo, conteúdo.

O pacote `graphicx` permite a inserção de gráficos no documento. Neste caso o ficheiro *exemplo2.dvi* não incluirá os gráficos, mas sim uma série de comandos para que o `dvips` (a opção usada neste caso) insira os gráficos. O pacote `graphicx` define um comando `includegraphics`, que é usado no exemplo 2 para incluir um gráfico PostScript (*disco.ps*) como se mostra na figura 5.2.

O comando `maketitle` cria o título do documento, com a informação definida pelos comandos `author`, `title` e `date`; neste caso não foi usado o comando `date` para definir a data, e consequentemente será usada a data do dia em que seja processado o ficheiro *exemplo2.tex*, a qual aparecerá em português, devido ao uso do pacote `babel`. Cá podemos apreciar a filosofia subjacente no LaTeX: o autor não tem que se preocupar com os pormenores técnicos de como será escrito o título (tamanho e tipo de letra, centrado, etc), mas sim com a sua estrutura lógica (o autor define os campos `author` e `title` e o `maketitle` trata do resto). Se estivessemos a produzir directamente um ficheiro como *exemplo2.ps* na figura 5.2, sem produzir o ficheiro *exemplo2.tex*, um programa de pesquisa não conseguia descobrir que parte do título representa o autor ou título, com as consequentes desvantagens.

Para criar listas com numeração automática, usa-se o **ambiente** enumerate; como qualquer outro ambiente, este deve ter um comando `begin` e um comando `end` que o delimitam. O início de cada alínea na lista é indicado por meio do comando `item`; o início de nova linha e a linha em branco antes de cada comando `item` no ficheiro *exemplo2.tex* é simplesmente para facilitar a sua leitura, não tendo nenhuma influência no aspecto final da lista. No exemplo 2 a primeira lista tem apenas uma alínea, e dentro dela foi criada uma outra lista que passou a usar uma numeração diferente, nomeadamente *a*, *b*,.... Se preferir que as alíneas de alguma das listas apresentem um círculo preto em vez de ser numerada, basta trocar o nome do ambiente `enumerate` por `itemize` no `begin` e `end` da respectiva lista.

Como já tinhamos referido, o símbolo $ é usado para escrever expressões matemáticas. Dentro das expressões matemáticas o LaTeX usa uma fonte diferente, semelhante à itálica mas com diferentes regras de espaçamento e com muitos símbolos matemáticos disponíveis. Outra forma de introduzir uma expressão matemática

é o ambiente `equation`; nesse caso a expressão matemática aparecerá centrada e numa linha separada, e com um número no lado direito que identifica a equação e que é actualizado automáticamente pelo `latex`. Dentro das expressões matemáticas podem ser usados subíndices e superíndices por meio dos símbolos _ e ^; o caracter imediatamente a seguir ao símbolo será o subíndice ou superíndice. Para escrever índices com mais do que um caracter será preciso usar chaves como por exemplo: `10^{12}`, o qual produzirá 10^{12}. O comando `frac` é usado dentro do modo matemático para produzir fracções; este comando tem dois operandos (numerador e denominador). O comando `\;` é usado no modo matemático para introduzir um pequeno espaço.

Outro ambiente usado no exemplo 2 é o ambiente `center` para centrar texto ou qualquer outro objecto. O comando `textbf` define texto em caracteres **negros**, e o comando `emph` define caracteres *itálicos*.

5.4 Gráficos vectoriais usando `xfig`

Uma ferramenta útil para desenhar diagramas para um documento é o programa `xfig`. O programa `xfig` cria gráficos **vectoriais**[2], num formato próprio e com nomes de ficheiro com terminação *.fig*; o programa permite também exportar o gráfico para outros formatos vectoriais ou *bitmap*, ou para uma série de comandos do LaTeX que podem ser introduzidos directamente num ficheiro LaTeX.

O primeiro passo que deverá ser feito é abrir o `xfig` com algumas opções para reduzir o seu tamanho, já que tal como vem configurado será impossível de visualizar a janela completa, a menos que trabalhe com um ecrã de alta resolução (1280×1024 ou superior). Umas opções apropriadas para resoluções gráficas de 800×600 ou 1024×768 são as seguintes

```
xfig -but_per_row 3 -pheight 15 -pwidth 20 &
```

A opção `but_per_row` aumenta a largura da área de botões na esquerda (figura 5.3), e as opções `pheight` e `pwidth` modificam as dimensões (em centímetros) da área de trabalho à direita. O `xfig` pode ter sido compilado com o uso de polegadas em vez de centímetros, e nesse caso será necessária a opção `metric` para mudar para centímetros.

A figura 5.3 mostra a janela do `xfig`, quando for executado com as opções sugeridas acima. A janela está dividida em 4 secções. A primeira secção é a barra horizontal superior com menus, uma linha com informação sobre o objecto que se está a desenhar, e um quadro à direita que mostra os três botões do rato e a sua

[2]Gráficos nos quais cada curva ou área são definidas em forma matemática em função de coordenadas de pontos em duas dimensões. A qualidade dos gráficos vectoriais permanece igual quando a figura for ampliada ou reduzida, o qual não acontece nos gráficos **bitmap**.

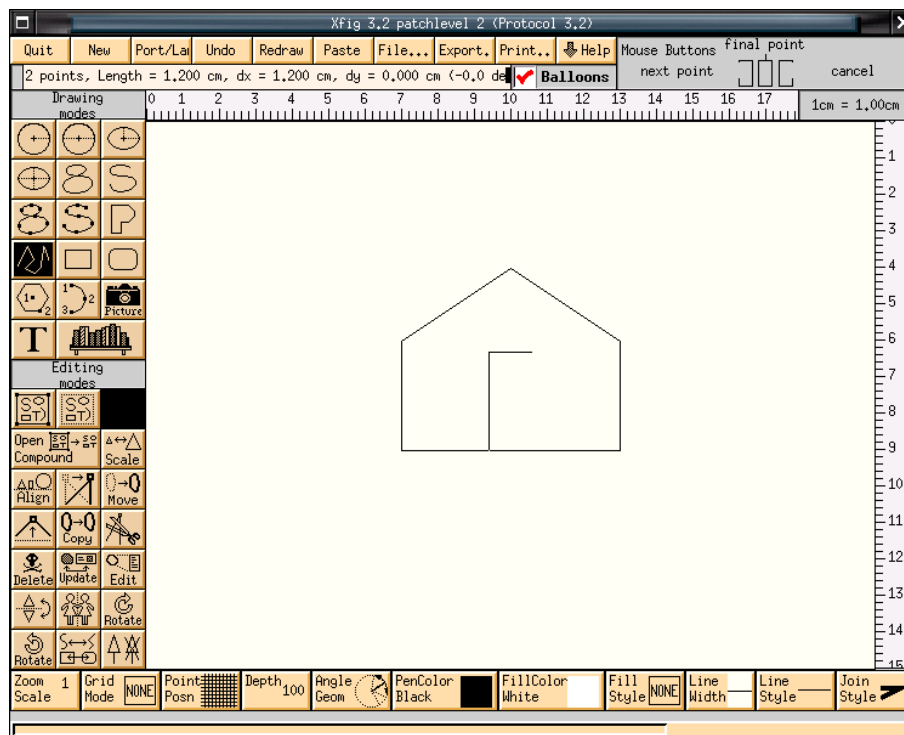



Figura 5.3: O programa `xfig`.

função. A segunda secção, à esquerda, são os botões para selecção de modos; os modos estão divididos em modos de desenho (*drawing modes*), usados para desenhar curvas ou polígonos, escrever texto, ou incluir imagens externas ou imagens existentes na biblioteca de imagens do `xfig`, e modos de edição (*editing modes*) que permitem combinar objectos para formar um objecto composto, ou modificar os objectos já existentes. A terceira secção é a área de trabalho à direita dos botões, e a quarta área é a zona de opções do modo que estiver a ser usado.

No exemplo da figura, está a ser usado o modo «PolyLine» para desenhar linhas quebradas, o qual aparece sublinhado na área de botões de desenho. A linha de informação por baixo do menu indica que têm sido usados 2 pontos na última linha polígonal desenhada (parte da porta da casa) e que o ponto onde se encontra o cursor está deslocado 1,2 cm na horizontal e 0 cm na vertical, em relação ao ponto anterior (ponto número 2); é de salientar que os deslocamentos verticais são medidos de cima para baixo no `xfig`. É importante também observar a informação no quadro com os três botões do rato; este indica que se carregarmos no botão esquerdo, fixaremos a posição do terceiro ponto, passando para o quarto ponto, se carregarmos no botão do centro finalizaremos a linha poligonal que estamos a desenhar, e se carregarmos no botão direito eliminaremos a linha poligonal que estavamos a desenhar, podendo começar uma nova. A outra área que é importante observar, antes de começar a desenhar um objecto, é a secção de opções por baixo da área de trabalho. Na figura 5.3 aparecem as opções com que está a ser usado o modo «PolyLine»:

espessura e estilo da linha, resolução da posição onde pode ser descolado o cursor, etc. Carregando duas vezes no botão de alguma opção, aprece um menu que permite a sua modificação.

Para aprender a usar o `xfig` é preciso experimentar bastante com os seus modos e opções. Se tiver seleccionada a opção de «Balloons», na parte superior da janela, quando deslocar o rato por cima de um botão ou menu obterá alguma informação sobre a sua função e, no caso dos menus, aparecerá indicado o uso dos diferentes botões do rato.

5.5 Outras fontes de informação

Neste capítulo temos falado de uma pequena parte das potencialidades do LaTeX que esperamos sejam suficientes para começar a usá-lo. Para obter uma visão mais completa, existem vários livros sobre o Latex; alguns deles são livres e podem ser obtidos em fomato tex, dvi, ps ou pdf. As distribuições de Tex/Latex costumam vir com bastante documentação; na distribuição Debian, a documentação encontra-se no directório */usr/doc/latex*, o qual contém também páginas em `html` com apontadores para a documentação.

Sendo o LaTeX um esforço conjunto de muitos voluntários existe também uma rede de servidores dedicados exclusivamente a receber as contribuições do público (pacotes e documentação), designados de **Arquivos CTAN** (*Comprehensive TeX Archive Network*); existem vários servidores da CTAN espalhados pelo mundo, com o mesmo conteúdo, embora os métodos de acesso possam ser diferentes. Alguns nodos da CTAN na Europa são os seguintes

```
http://ftp.rediris.es/
http://ftp.tex.ac.uk/
http://ftp.dante.de/
http://ftp.univie.ac.at/
```

Grande parte da documentação incluída com programas para Linux tem sido feita usando LaTeX; consequentemente, uma boa fonte para aprender LaTeX consiste em estudar os ficheiros fonte de algum manual que tenha alguma característica que deseje imitar nos seus documentos. O rápido crescimento de sistemas livres como o LaTeX é devido à liberdade de distribuir e copiar informação entre autores e utilizadores.

Apêndice A

Como foi feito este documento

A maneira de exemplo da criação de um documento em LaTeX, apresentaremos neste apêndice o ficheiro principal usado para criar estes apontamentos. O trabalho foi desenvolvido, em simultâneo, por vários colaboradores; foi usado o sistema RCS (*Revision Control System*) que guarda uma história e uma cópia das diversas versões, e permite impedir que um autor modifique um ficheiro em quanto outro autor não terminar as suas modificações. O documento foi dividido em vários ficheiros para permitir que dois autores pudessem modificar diferentes partes sem interferir, e foi usado um ficheiro *guia.tex* que define o estilo geral do documento e permite processar uma ou várias partes do documento. O ficheiro *guia.tex* é o seguinte:

exemplo1.tex

```
\documentclass[a4paper, 12pt]{report}

% O seguinte if permite descobrir se esta a ser
% usado pdflatex ou latex o qual determinará o uso
% de dvips ou pdftex para produzir o ficheiro final

\newif\ifpdf\ifx\pdfoutput\undefined\pdffalse
\else\pdfoutput=1\pdftrue\fi
\ifpdf\newcommand{\driver}{pdftex}
\else\newcommand{\driver}{dvips}\fi

% Definição da lingua e alfabeto a usar
\usepackage[latin1]{inputenc}
\usepackage[portuges]{babel}
\usepackage[T1]{fontenc}

% Pacotes gráficos (\driver já foi definido acima)
\usepackage[\driver]{color}
\usepackage[\driver]{graphicx}
```

```
\usepackage{times}    % fontes Times da Adobe em
                      % vez das ComputerModern do tex
\usepackage{ifthen}   % ’if’ será usado para decidir
                      % quais capítulos processar
\usepackage{fancyhdr} % uns cabeçalhos mais
                      % engraçados que os normais.
\usepackage{fancyvrb} % um ambiente ’verbatim’ com
                      % mais opções.

% Definição das dimensões da página
\setlength{\textwidth}{14cm}
\setlength{\textheight}{22cm}
\setlength{\oddsidemargin}{1cm}
\setlength{\evensidemargin}{0pt}

% Formato dos cabeçalhos, usando ’fancyhdr’
\pagestyle{fancyplain}
\renewcommand{\chaptermark}[1]{\markboth{#1}{}}
\renewcommand{\sectionmark}[1]
                {\markright{\thesection{} #1}}
\lhead[\fancyplain{}{\bfseries\thepage}]
    {\fancyplain{}{\itshape\rightmark}}
\rhead[\fancyplain{}{\itshape\leftmark}]
    {\fancyplain{}{\bfseries\thepage}}
\cfoot[]{}

% Algumas cores para facilitar a leitura dos
% exemplos em LaTeX
\definecolor{violet}{rgb}{0.8,0.2,0.5}
\definecolor{brown}{rgb}{0.6,0.5,0.1}
\newcommand{\red}[1]{\textcolor{red}{#1}}
\newcommand{\blue}[1]{\textcolor{blue}{#1}}
\newcommand{\green}[1]{\textcolor{green}{#1}}
\newcommand{\brown}[1]{\textcolor{brown}{#1}}
\newcommand{\violet}[1]{\textcolor{violet}{#1}}
\newcommand{\backsl}{\ensuremath{\backslash}}

% Para as imagens bitmap usaremos formato png, mas
% para produzir um ficheiro PostScript usando
% dvips as imagens será fornecidas já em ps.
\newcommand{\fig}[1]{\ifpdf\includegraphics{#1.png}
  \else\includegraphics{#1.ps}\fi}
```

```
% Para figuras vectoriais usamos pdf ou ps
%  conforme \driver for pdftex ou dvips
\newcommand{\vecfig}[1]
 {\ifpdf\includegraphics{#1.pdf}
 \else\includegraphics{#1.ps}\fi}

% para nomes de comandos Unix:
\newcommand{\nome}[1]{\texttt{#1}}
% nomes de ficheiros:
\newcommand{\fich}[1]{\emph{#1}}
% palavras em lingua estrangeira:
\newcommand{\est}[1]{\emph{#1}}
% comandos Unix (sinónimo de nome):
\newcommand{\comando}[1]{\texttt{#1}}
% algum conceito novo que seja explicado a seguir:
\newcommand{\definicao}[1]{\textbf{#1}}

% A informação do RCS ficará no ficheiro .log
\typeout{Id: guia.tex,v 1.11 1999/09/28 14:11:29}

% Esta parte do código permite processar um ou
% vários capítulos
\typein[\chaps]{ESCOLHA:capa, intro, cap1, cap2,
  cap3, apendices (ou todos)}
\ifthenelse{\equal{\chaps}{todos}}{}
    {\includeonly{\chaps}}

\begin{document}
\include{capa}
\include{intro}
\include{cap1}
\include{cap2}
\include{cap3}
\appendix
\include{apendices}
\end{document}
```

Apêndice B

Glossário de Informática e Redes

Neste apêndice aparece uma lista parcial de termos de informática. Alguns destes termos foram usados neste manual, e outros não, mas são incluídos cá por aparecerem frequentemente nos temas relacionados com o manual.

A

API (Application Programming Interface). Rotinas usadas para requisitar acções de baixo nível ao sistema operativo. Usado no desenvolvimento de aplicações.

APM (Advanced Power Management). Recurso para economia de energia.

ASCII. tabela de símbolos, usada para gerar e armazenar os códigos que não podem ser criados via teclado. Foi definida pela ANSI.

Auto-estrada da informação / auto-estrada electrónica. Uma ligação ou conjunto de ligações entre computadores, formando uma rede de redes, de preferência com meios de comunicação extremamente rápidos. Um nome abusivamente usado por vezes (sobretudo nos media tradicionais) para designar a(s) rede(s) actualmente existente(s) (e em particular a Internet), pois uma grande parte delas ainda tem muitas ligações bastante lentas.

B

Backup. cópia de segurança dos ficheiros de um computador.

Bauds (ou bps). velocidade de comunicação (portas série ou portas paralelas). Exemplo: modem de 28.800 bauds.

BBS (Bouletin Board System). sistema de comunicação, usado para comunicação e troca de informações. Foi substituído em grande parte pela Internet, que possui melhor operacionalidade.

BIOS (Basic Input Output System). sistema existente nos PC's que permite a configuração e armazenamento das características de hardware do computador.

Bit. quantidade mínima de informação. Genericamente é representado por 0 ou 1.

Boot. inicialização do computador.

Boot record. registo de inicialização, parte do disco rígido que contém a inicialização do sistema operativo.

Byte. a unidade de medida da informação (8 bits). Equivale basicamente a um caracter.

Browser. programa que ajuda na navegação pela WWW. Um browser ajuda o utilizador a encontrar e obter a informação desejada. O Netscape Navigator e o Internet Explorer são os browsers mais comuns.

Buffer. memória intermediária auxiliar, que facilita as tarefas, pois liberta o processador.

Bug. erro escondido num programa (erro de programação).

Bus. condutores eléctricos dentro do computador (estradas por onde os dados seguem).

C

Cache. memória auxiliar que facilita a recuperação de informações recentemente/frequentemente acedidas.

CAD / Computer Aided Design (Projecto Assistido por Computador) programas usados para projectos de engenharia, de arquitectura e científicos — construção de automóveis, aviões, edifícios, etc.

CD-ROM. discos compactos laser, que contêm informação (dados, sons, música) que podem ser lidos, mas não alterados e nem re-gravados.

Chip. designação de circuito integrado, que inclui desde dezenas a milhares de circuitos electrónicos em si.

Ciberespaço. por ciberespaço designa-se habitualmente o conjunto das redes de computadores interligadas e de toda a actividade aí existente. É uma espécie de planeta virtual, onde as pessoas (a sociedade da informação) se relacionam virtualmente, por meios electrónicos. Termo inventado por William Gibson no seu romance Neuromancer.

Client / cliente / estação de trabalho. computador que acede aos recursos partilhados de rede fornecidos por outro computador.

CMYK. sistema de cores usado para impressão — qualquer tom de cor pode ser gerado usando a combinação CMYK (ciano, magenta, amarelo e preto).

Compactação / Compressão. compactar os dados para ocupar menos espaço de armazenamento ou menos tempo para o envio/recepção do arquivo. Os programas mais usados em Linux para compactar e des-compactar são gzip, zip, unzip e compress.

Compilador. programa que transforma uma lista de comandos em forma de textos (legível) para a forma de linguagem de máquina, que somente o computador identifica e usa para execução. O computador não executa directamente os comandos em forma de texto.

Correio electrónico. ver e-mail.

CPU. parte principal do computador (processador e registos internos).

Criptografia. sistema de segurança, que transforma matematicamente um ficheiro em símbolos, que apenas poderá ser de-codificado por alguém que tiver a tabela ou fórmula de de-criptografia específica para aquele ficheiro.

Cursor. sinal no écran (geralmente estático ou a piscar) que nos indica onde no écran deve ser inserida a próxima informação (pode ser alterado com o uso do teclado ou do rato).

D

Default / valor padrão. um programa pede uma informação e oferece uma resposta padrão (mais comum). Se não altera, assume este valor default.

Desktop. área de trabalho do écran — espaço visual do écran.

DMA (direct memory access) / acesso directo à memória. a informação é transferida entre a memória e o disco rígido, sem passar pelo processador — o mais rápido. Usado por alguns aplicativos.

Domínio. nome de uma rede de computadores com a última parte do endereço IP comum. Ao último termo de um domínio chama-se domínio de topo. Cada país tem um domínio de topo. Por exemplo:

- au - Austrália
- ca - Canadá

- de - Alemanha
- es - Espanha
- fr - França
- nl - Holanda
- no - Noruega
- pt - Portugal
- se - Suécia
- uk - Reino Unido
- mz - Moçambique
- br - Brasil

Se o domínio de topo não for um código de um país, é bastante provável que o país seja os E.U.A. (embora hoje em dia outros países tenham começado a usar os mesmos domínios). Os domínios de topo americanos são:

- com - organizações comerciais
- edu - instituições de ensino / educação
- gov - organizações governamentais
- mil - organizações militares
- net - fornecedores de serviços Internet
- org - organizações sem fins lucrativos

Domínio público. algo que está no domínio público (software, p.ex.) é algo que se pode copiar, cortar, colar, queimar, distribuir, deitar ao lixo e nomeadamente utilizar sem pagar o que quer que seja! Não confundir com o *Software* Livre! (ver *Freeware*.)

Driver. programa (software) que controla um dispositivo de hardware.

E

EISA (Enhanced Industry Standard Architecture). padrão de barramento interno dos computadores, superior ao ISA.

E-mail / electronic mail / correio electrónico. correio transmitido por meios electrónicos, normalmente redes informáticas. Uma carta electrónica contém texto (como qualquer outra carta) e pode ter sons, imagens ou outros documentos anexos.

Endereço. ver site.

EPS (Encapsulated PostScript). Ver PostScript.

F

FAQ (Frequently Asked Questions) / Perguntas Feitas Frequentemente solução para as dúvidas mais comuns (relação de perguntas e respostas já feitas).

Fila de impressão / spool. programa que controla a fila de impressão. Quem enviou tarefas para imprimir primeiro, prioridades, etc.

Fontes. tipos de letras instalados no computador.

Formatar. dividir um disco rígido ou flexível em sectores, para possibilitar o armazenamento das informações.

Fórum de discussão. ver newsgroup.

Freeware. termo que costuma ser mal interpretado, devido ao duplo sentido da palavra *Free*. Neste caso *Free* deve ser traduzida como livre e não como grátis. O *Software Livre* pode ser vendido ou distribuído gratuitamente, modificado e redistribuído, mas em todos os casos existem leis que o protegem para impedir que alguém tome controlo sobre ele. Nomeadamente, se for vendido por uma companhia, esta está obrigada a distribuir também o código fonte que permite a qualquer outro modificá-lo, copiar e até re-vender. Um programa livre não deve ser confundido com programas do domínio público, pois no caso do domínio público alguém pode fazer com que as fontes desapareçam e continuar a distribuir binários sendo o único em poder das fontes; isso seria ilegal no caso do *Freeware*. O Linux e a maior parte dos programas distribuídos com ele são *Freeware*.

FTP (File Transfer Protocol). método de transferência de dados entre computadores.

Full-duplex. quando a informação circula em ambas as direcções pela linha de dados. Por exemplo: ao conversar na Internet, se tivermos uma placa de som full-duplex poderemos falar e ouvir ao mesmo tempo.

G

Gateway. porta de comunicação.

GIF. formato proprietário para ficheiros de formato gráfico. Por usar unicamente 8 bits por ponto e permitir compactação, produz ficheiros de pequenas dimensões, mas não permite usar mais do que 256 cores.

Grupo de news. ver newsgroup.

GUI (Graphical User Interface). maneira gráfica de lidar com os aplicativos. O X é uma interface gráfica.

H

Hard Disk. disco rígido interno do computador usado para armazenar permanentemente as informações.

Hardware. conjunto de material que constitui ou está integrado num dado computador, é a sua parte física.

Hidden / escondido. ficheiro que existe fisicamente, mas não está visível; em Unix/Linux o primeiro caracter no seu nome deverá ser um ponto.

Homepage. pode ser a página web carregada automaticamente pelo browser quando arranca, ou ser a página principal de um conjunto de páginas dedicadas a um assunto. Costuma conter uma mistura de gráficos e texto, assim como referências (hyperlinks) para páginas com ela relacionadas.

Host. ver servidor.

Hipertexto (hypertext). Formato que permite que a informação seja organizada num formato não sequencial, e acessível através de ligações directas (links). Deste modo, o hypertexto permite aos utilizadores o acesso à informação a partir de múltiplos pontos de entrada. Dois sistemas de hipertexto usados em Linux são a linguagem `html` e o sistema `info`.

I

Ícone. símbolo gráfico (pequena figura) que é uma maneira rápida de executar uma acção (exemplo, clique no ícone da impressora para imprimir).

IDE. sistema de troca de dados do disco rígido para a placa controladora do microcomputador, o mais usado. Existem sistemas mais rápidos, como o Fast-IDE (Fast-ATA) ou SCSI (tecnologia bem diferente, voltada para multiprocessamento - redes, ou scanners, CD ROMS).

Importar. trazer dados, informações de outro programa.

internet. com i minúsculo, internet designa apenas uma rede de redes e não específicamente a Internet.

Internet. a melhor demonstração real do que é uma auto-estrada da informação. A Internet (com I maiúsculo) é uma imensa rede de redes que se estende por todo o planeta e practicamente todos os países, possibilitando diversos tipos de serviços informáticos.

ISP (Internet Service Provider). fornecedor de serviços de internet — a empresa que nos fornece acesso à Internet.

Intranet. rede interna de informações numa empresa, que usa o mesmo formato de armazenamento e busca de informações da Internet. Usada como forma de comunicação e divulgação de informações internas numa empresa.

IP / endereço IP. endereço Internet Protocol, usado para especificar as rotas de comunicação.

IRQ/interrupção (Interrupt Request Lines). endereço físico no hardware do computador. Alguns dispositivos/placas internas do computador usam números de interrupção para indicar ao processador o seu estado. Dentro do microcomputador, geralmente, cada dispositivo deve ter uma interrupção diferente, para que não haja conflitos.

ISO (International Organization for Standardization). organização que estabelece normas de qualidade a nível mundial. Normas de procedimento/padronização.

J

JAVA. linguagem de programação que também pode ser utilizada para produzir páginas Internet, com mais recursos que a linguagem HTML.

JPEG, JPG. extensão para ficheiros de formato gráfico, em que os ficheiros podem ser compactados a vários níveis, chegando a ocupar espaço muito reduzido; parte da informação é perdida conforme o grau de compactação, mas nas imagens fotográficas a qualidade é possível manter uma alta qualidade com alto grau de compactação.

Jumper. pequenos interruptores de plástico usados nas placas de hardware. Por ou tirar um jumper, significa mudar a operação das placas. Sistema de configuração das placas.

K

Kernel. a parte do sistema operativo que gere o processador.

Keyboard. teclado.

L

LAN (Local Area Network). ligação de computadores em rede, através de cabos físicos.

Laptop. computador portátil. Os notebooks e subnotebooks são ainda menores que os laptops.

LCD (Liquid Crystal Display). écran de cristal líquido utilizado em alguns modelos de computadores portáteis.

Link. na Web, uma palavra destacada indica a existência de um link, que é uma espécie de apontador para outra fonte de informação, uma espécie de elo de ligação. Escolhendo esse link, obtém-se a página de informação que ele designava que pode, por sua vez, ter também vários links.

Linux. Sistema operativo livre, criado por Linus Torvalds mas actualmente desenvolvido por um grupo aberto de voluntários no mundo inteiro. Inicialmente criado como versão compatível do Unix para computadores PC, mas hoje em dia existem versões para muitas plataformas e workstations. (ver *Freeware*.)

Login / Logon. identificação de um utilizador perante um computador. Fazer o login (ou logon) é o acto de dar a identificação de utilizador ao computador.

Logout. acto de desconectar a sua ligação a um determinado sistema ou computador, finalização do acesso ao sistema.

M

Macro. sequência de comandos (pequeno programa) para automatizar procedimentos.

Máquina de pesquisa. ver search engine.

MBR (Master Boot record) / registo de inicialização. parte do disco rígido que contém a inicialização do sistema operativo. Em Linux o MBR pode ser alterado por meio do programa `lilo`, para dar a possibilidade de arrancar o computador em diferentes sistemas operativos.

Modem (modulador/des-modulador de sinais). aparelho que permite comunicação entre computadores.

Motores de procura. ver search engine.

Mouse / Rato. dispositivo que auxilia no manuseio do sistema, principalmente em sistemas gráficos. O movimento que faz com o rato, é reflectido no écran. Indispensável auxiliar do teclado.

MS-DOS. sistema operativo da Microsoft, prévio ao aparecimento do Windows 95.

Multimédia. presença de vários tipos de informação (texto, imagens, audio, vídeo, etc.) no mesmo suporte.

N

Navegar. na Internet significa vaguear, passear, procurar informação, sobretudo na Web.

net. ver rede.

Net. Net (com N maiúsculo) é uma abreviatura para designar a Internet.

Netiquette. conjunto de regras e conselhos para uma boa utilização da rede Internet, de modo a se evitarem erros próprios de novatos aquando da interacção com outros utilizadores (mais experientes). A netiquette baseia-se muito no simples e elementar bom senso.

Netscape. nome de um dos mais comuns browsers da Internet, é também o nome da empresa que o fabrica.

Network. chama-se network sempre que se interligam dois computadores ou mais de forma a constituirem uma rede e portante capazes de partilhar recursos. Ligando duas ou mais networks forma-se uma internet (com inicial minúscula).

News. ver newsgroup.

Newsgroup. são grupos de discussão, organizados por temas, onde se pode ler e escrever (publicamente) sobre o tema indicado pelo nome do grupo de discussão.

O

Offline. quando o computador não tem ligação à rede, está offline.

Online. por oposição a offline, online significa "estar em linha", estar ligado em determinado momento à rede ou a um outro computador. Para alguém, "estar online", significa que nesse momento essa pessoa esteja a usar a Internet e que tenha, portanto, efectuado o login num determinado computador da rede.

P

Página WEB ou WWW. ver homepage.

Palavra-chave. ver password.

Parâmetro. uma variável que é passada para um programa ou rotina (parte de um programa).

Partição. divisão lógica de um disco rígido, de forma que pareçam existir várias unidades físicas distintas. Em Linux o primeiro disco mestre encontra-se dividido nas partições */dev/hda1*, */dev/hda2*,..., o primeiro disco escravo tem as partições */dev/hdb1*, */dev/hdb2*,..., etc.

Password. código usado para aceder a um sistema fechado, senha de acesso.

PC (Personal Computer). computador pessoal; termo geralmente usado para computadores com a arquitectura dos modelos IBM-PC, baseados em micro-processadores Intel 286/386/486/Pentium, e compatíveis.

PCI (Peripheral Component Interconnect). barramento que sucedeu ao Vesa Local Bus. Utilizado na maioria dos computadores Pentium.

PCMCIA (Peripheral Computer Memory Card International Association) dispositivos de hardware em forma de pequenos cartões. Em geral, usado para computação portátil (laptops e notebooks). Exemplo: cartões PCMCIA de memória, fax/modem, de rede, etc.

PDF (Portable Document Format). formato usado para representar documentos; um ficheiro pdf pode conter várias páginas, e cada página pode ter uma combinação de texto e imagens com um formato que será independente do dispositivo e resolução usados para imprimir ou visualizar. Também permite a criação de hipertextos.

Pentium. processador mais avançado, actualmente, para a linha de PC's.

Pixel. o menor ponto do écran o dispositivo gráfico. O écran é dividido em milhares de pontos. A resolução do écran pode ser configurado pelo utilizador.

Plataforma. o tipo de computador, processador, sistema operativo em uso. Exemplo: Este programa funciona somente em plataforma PC-Linux. Ou, funciona em plataforma Macintosh, RISC-Unix, PC-MSWindows, SPARC-Linux, etc.

PNG. formato gráfico criado para substituir ao `gif`. Permite também diferentes graus de compactação sem perda da qualidade, mas usa um algoritmo de compactação livre de patentes, e permite usar muitas mais cores do que as 256 do formato `gif`.

Portas série. portas usadas para ligar ratos, modems, etc. Encontram-se normalmente em */dev/ttyS0*, ..., */dev/ttyS3*. Os bytes são enviados em fila pela porta série, uns atrás dos outros.

Portas paralelas. portas usadas para ligar impressoras, algumas câmaras de vídeo, etc. Encontram-se normalmente em */dev/lp0*, ..., */dev/lp1*. Os bytes são enviados de 8 em 8 (em geral) pela porta paralela, tornando a comunicação paralela bem mais rápida que a série.

PostScript. Linguagem da Adobe para criação de gráficos vectoriais, inicialmente para impressoras, que se tem convertido num padrão. O programa livre *GhostScript* é uma implementação do *PostScript* com algumas extensões.

PPP (Point-to-Point Protocol). protocolo (método) usado para comunicação entre computadores, em geral na Internet e usando modem.

Proxy. computador intermediário numa ligação, recebe os pedidos de uma rede e dá-lhes seguimento.

Public domain. ver domínio público.

Q

Query. Consulta / pesquisa.

Quit / sair. finalizar o uso de um programa.

R

RAM (Random Access Memory). memória principal temporária do computador. É esvaziada quando desligamos o computador.

Realidade virtual. simulação do mundo real (visão, som, sensações tácteis, etc.) por processos inteiramente controlados através de meios electrónicos.

Rede. conjunto de computadores ligados por forma a serem capazes de comunicar entre si.

Reset/ Reboot. reinicializar o computador. A combinação de teclas: Ctrl+Alt+Del pode estar configurada para fazer o *Reboot* ou o *Shutdown*. O botão de reset não deve ser usado em Linux, sem antes ter feito um *Shutdown*. (ver *Shutdown*)

Resolução. a qualidade da placa de vídeo, do monitor de vídeo ou da impressora, quanto ao número de pontos (pixels) que pode apresentar.

ROM (Read Only Memory) / Memória Só de Leitura. memória não volátil (permanece quando computador é desligado). Só de leitura, não pode ser alterada pelo utilizador. A BIOS é gravada em ROM. Os dados configuráveis da BIOS em EPROM.

Router. computador, software ou material dedicado que serve para interligar duas ou mais redes efectuando automaticamente a re-direcção correcta dos pacotes de informação de uma rede para outra.

RS-232. sistema de comunicação série.

S

Scanner. dispositivo que transporta imagens do papel para a memória do computador.

Script. conjunto de comandos ou programa que é interpretado sem ter que ser compilado.

Search engine. ferramenta de software usada na pesquisa de servidores na Internet com páginas relativas a um determinado tema ou que incluiam determinadas palavras.

Server / Servidor. o computador que administra e disponibiliza recursos na rede, partilhando software e os seus ficheiros com os utilizadores, que acedem a partir das estações ou de seus computadores remotos.

Sector. divisão (gavetas) de armazenamento nos discos (rígidos e flexíveis).

Shareware. software que é distribuído gratuitamente mas não livremente (o código fonte não é distribuído). Normalmente, foi feito para ser testado durante um curto período de tempo e, caso seja utilizado, o utilizador deve registá-lo.

Shell. A interface do sistema operativo.

Shutdown. finalizar todas as aplicações, gravar os dados que estão na memória a fim de poder desligar com segurança o computador. Nunca desligue o computador sem ter feito primeiro o shutdown; os programas e dados activos na memória, serão perdidos e o disco rígido pode ser danificado. A combinação de teclas: Ctrl+Alt+Del pode estar configurada para fazer o *Shutdown* ou o *Reboot*. (ver *Reset*.)

Sintaxe. forma e ordem na qual os comandos e elementos devem ser digitados.

Sistema Operativo. sistema (programa) que faz comunicação entre o hardware e os demais softwares. Exemplo: Linux, Windows95, WindowsNT, etc.

Site. local; diz-se da página de uma empresa ou pessoa. É um dos computadores da Internet ou o caminho até uma das páginas existentes nesse computador.

SLIP (Serial Line IP). protocolo (método) usado para comunicação entre computadores usando a porta série, em geral na Internet.

Smile / smiley. são pequenos conjuntos de caracteres utilizados em correio electrónico e foruns de discussão que pretendem transmitir uma emoção ou estado de espírito. Os mais conhecidos são:

- `:-) ou :)` - sorriso, estou a brincar.
- `;-) ou ;)` - piscar de olho.
- `:-( ou :(` - triste, carrancudo.
- `:-*` - beijo.
- `:-P ou :P` - caretas (pôr língua de fora).

SMTP (Simple Mail Transport Protocol). protocolo de transmissão de ficheiros via correio electrónico.

Snail mail. referência irónica ao correio postar (snail = caracol), comparando-o com a celeridade do e-mail.

Software. conjunto de programas que corre, ou está preparado para correr, num dado computador.

Software Livre. ver *Freeware*.

Spam. publicação do mesmo artigo de news em vários grupos de discussão ou envio exagerado de vários mails, geralmente para utilizadores que não o solicitaram, geralmente resultando em desperdício de espaço em disco e largura de banda nos meios de transmissão.

Spooler. programa que controla a fila de impressão. Quem enviou tarefas para imprimir primeiro, prioridades, etc.

Swap. Área de um disco rígido que é destinada à emulação da memória RAM. Uma área de swap permite a um sistema funcionar como se tivesse mais memória RAM, mas o seu uso torna o sistema mais lento, pois o processador que transferir transferir partes da memória swap para a memória RAM física cada vez que precisar ter acesso a ela.

T

TCP/IP (Transmission Control Protocol / Internet Protocol). protocolo (método) de comunicação entre computadores.

Telnet. programa que permite trabalhar num computador remoto ligado à Internet como se estivéssemos frente a um terminal fisicamente ligado a esse computador.

Tempo médio de acesso. termo que se refere ao tempo de resposta médio que um dispositivo leva para aceder à informação. Exemplo: memória de 60 ns , disco rígido de 13 ms.

Time-out. sinal que um dispositivo emite quando chegou o tempo limite no qual estava à espera de receber uma informação.

Topologia. disposição e modo de ligação dos computadores numa rede (em estrela, em anel, etc.).

U

Unix. sistema operativo com características de multi-tarefa; os primeiros computadores ligados à Internet usavam este sistema exclusivamente. O Linux é um sistema operativo criado para ser compatível com o Unix, mas que por não ser comercial não tem obtido o selo de aprovação da empresa que mantém os direitos sobre a marca registada Unix.

UPS (Uninterruptible Power Supply). fonte de alimentação ininterrupta; sistema com baterias, que mantém o computador a funcionar por um determinado período.

URL (Uniform Resource Locator). ver site.

USB (Universal Serial Bus). sistema de comunicação série, que permite a ligação de diversos periféricos.

Username. ver login.

V

Virtual Reality. ver realidade virtual.

W

Web. diminutivo para World Wide Web (à semelhança de WWW ou W3).

Webmaster. criador/responsável por páginas ou sites na Internet (páginas Web).

Windows (3.1, 95, NT etc). sistemas operativos da Microsoft.

World Wide Web. subconjunto da Internet, em que a informação é organizada em documentos multimédia interligados por hyperlinks.

WWW. ver Web.

WYSIWYG (What You See Is What Get). "O que vê é o que terá"; o que vê no écran, terá exactamente o mesmo aspecto na impressora.

X

X. Sistema gráfico de janelas que pode ser executado em plataformas Unix e compatíveis. Está dividido em duas partes: cliente e servidor, que podem ser dois computadores diferentes e inclusivamente com sistemas operativos diferentes. O servidor recebe a entrada dos programas do cliente e envia para eles a saída.

Y

Y2K. sigla referente ao problema de transição do ano 2000 nos sistemas informáticos, o "bug do ano 2000".

Z

Zip. Formato usado para criar arquivos de ficheiros comprimidos. Um arquivo Zip costuma ter um nome terminado em *.zip*. Podem ser criados usando o programa `zip`, e para extrair e descomprimir ficheiros pode ser usado o programa `unzip`.

Bibliografia

[1] Lamport, L. *LaTeX: A Document Preparation System. Addison-Wesley, U. S. A., 1986, ISBN: 0-201-15790-X.*

[2] Welsh, M. e L. Kaufman. *Running Linux*, O'Reilly & Associates, U. S. A., 1995, ISBN: 1-56952-100-3.